Handling bibliographies with texor

by Abhishek Ulayil

Abstract This is a small sample article to demonstrate usage of **rebib** to manage bibliographies.

1 Introduction

The **rebib** package is a spin-off from the **texor** package.

The spin-off cam about because the bibliography section in **texor** package was expanding significantly, enough to deserve a dedicated space.

Thus **rebib** was born and it has improved over the past few months to include more advanced features and utilities.

What does rebib do?

In LaTeX, we have multiple ways to declare bibliography entries, which include embedding entries formatted to a given style in the document itself or isolating that section to an external file commonly called a '.bbl' file. Then we have the BibTeX format, which is designed to store the full bibliographic information for each reference, making it superior for managing references.

LaTeX has no problem dealing with multiple formats and works well with all these bibliographic formats. However, some web publishing journals like R Markdown articles for the R Journal do not support embedded bibliographies. As a result, they need a BibTeX equivalent of the same.

The **rebib** package converts embedded LaTeX bibliographies into a close BibTeX equivalent. This package relieves the author from manual conversion of the documents and aids the project to convert legacy R Journal articles to the current R Markdown format.

Features of rebib

- Extracts embedded bibliographic entries from a '.tex' file.
- Creates the mandatory title and author fields.
- Creates the optional URL, ISBN, publisher, pages and year fields, when available.
- Stores the remaining information in "journal" (internally) and "publisher" (when writing BibTeX file).
- Ignores commented LaTeX code.
- Tracks citations included in the document.
- Aggregates embedded bibliographic entries with references from supplementary '.bbl' and '.bib' files.

2 Parser

The magic behind **rebib** is a parser which parses bibliographic data and sorts them according to the matching regular expressions.

Stage 1: Read the embedded bibliography block

This stage is a minor step where it reads the Embedded Bibliography from the LaTeX document. This step also includes filtering out commented lines to avoid reading un-intended entries.

The bibliography is split into entries using the LaTeX macro `\\bibitem` as a marker for a new entry and the resulting data is exported to a variable.

```
dir.create(your_article_folder <- file.path(tempdir(), "exampledir"))
example_files <-  system.file("article", package = "rebib")
x <- file.copy(from = example_files,to=your_article_folder,recursive = T)
your_article_path <- paste(your_article_folder,"article",sep="/")
file_name <- rebib:::get_texfile_name(your_article_path)
```

```
bib_items <- rebib:::extract_embeded_bib_items(your_article_path,file_name)
bib_items[[1]]
#> [1] "\\bibitem[Ihaka, Ross and Gentleman, Robert]{ihaka:1996}"
#> [2] "Ihaka, Ross and Gentleman, Robert"
#> [3] "\\newblock \\emph{R: A Language for Data Analysis and Graphics.}"
#> [4] "\\newblock \\emph{Journal of Computational and Graphical Statistics}, 3:\\penalty0"
#> [5] "299--314, 1996."
#> [6] "\\newblock URL : \\url{https://doi.org/10.1080/10618600.1996.10474713}"
```

Stage 2: Regex powered parser

Now, with the chunks of bibliographic entries, each is passed to a parser which will break it down based on regular expressions. The logic is to use the LaTeX macro \\newblock as a placeholder to identify the position of text elements relative to it.

The first value to be parsed is the unique_id, also called the citation reference which is used to cite elements inside the article. Usually, this is in the first or second line of the whole entry. The position of the unique_id will determine the position of the author names.

```
bib_items[[1]][1]
#> [1] "\\bibitem[Ihaka, Ross and Gentleman, Robert]{ihaka:1996}"
```

After reading the unique_id, the parser will attempt to read the author name(s).

```
bib_items[[1]][2]
#> [1] "Ihaka, Ross and Gentleman, Robert"
```

Next, the title is extracted based on the position of the new blocks or the end of the bib chunk.

```
bib_items[[1]][3]
#> [1] "\\newblock \\emph{R: A Language for Data Analysis and Graphics.}"
```

This way the crucial elements of the bibliographic entry (unique_id, author names and title) are parsed out. The remaining data is stored as journal internally and publisher when writing to a new BibTeX file.

```
bib_items[[1]][4:6]
#> [1] "\\newblock \\emph{Journal of Computational and Graphical Statistics}, 3:\\penalty0"
#> [2] "299--314, 1996."
#> [3] "\\newblock URL : \\url{https://doi.org/10.1080/10618600.1996.10474713}"
```

A series of filters for ISBN, URL, pages and year fields are applied to search for relevant data from the remaining data. If the data is not available, then it is set as NULL and will be skipped while writing the BibTeX file. There is a lot of filtering of common LaTeX elements and after that, the remaining data is stored in a structured format to be written to a file.

```
bib_entry <- rebib:::bib_handler(bib_items)
bib_entry
#> $book
#> $book[[1]]
#> $book[[1]]$unique_id
#> [1] "ihaka:1996"
#>
#> $book[[1]]$author
#> [1] "Ihaka, Ross and Gentleman, Robert"
#>
#> $book[[1]]$title
#> [1] "R: A Language for Data Analysis and Graphics"
#>
#> $book[[1]]$journal
#> [1] "Journal of Computational and Graphical Statistics 3:   :"
#>
#> $book[[1]]$year
#> [1] "1996"
#>
#> $book[[1]]$URL
#> [1] "https://doi.org/10.1080/10618600.1996.10474713"
#>
#> $book[[1]]$isbn
#> NULL
#>
#> $book[[1]]$pages
#> [1] "299--314"
#>
#>
#> $book[[2]]
#> $book[[2]]$unique_id
#> [1] "R"
#>
#> $book[[2]]$author
#> [1] "R Core Team"
#>
#> $book[[2]]$title
#> [1] "R: A Language and Environment for Statistical Computing"
#>
#> $book[[2]]$journal
#> [1] "R Foundation for Statistical Computing Vienna Austria    :"
#>
#> $book[[2]]$year
#> [1] "2016"
#>
#> $book[[2]]$URL
#> [1] "https://www.R-project.org/"
#>
#> $book[[2]]$isbn
#> [1] "3-900051-07-0"
#>
#> $book[[2]]$pages
#> NULL
```

Stage 3: BibTeX writer

After reading the bibliographic entries and splitting out meaningful values from them, we can finally write a structured file in the BibTeX format.

The writer will read the bib chunks one at a time based on the metadata extracted and will write the corresponding data fields. The default entry type is a book, which should not have any problems with the web articles.

```
file_path <- paste0(your_article_path,"/",file_name)
bib_path <- paste0(your_article_path,"/example.bib")
rebib:::bibtex_writer(bib_entry,file_path)
cat(readLines(paste(your_article_path,"example.bib",sep="/")),sep = "\n")
#> @book{ihaka:1996,
#> author = {{Ihaka, Ross and Gentleman, Robert}},
#> title = {{R: A Language for Data Analysis and Graphics}},
#> publisher = {Journal of Computational and Graphical Statistics 3:   :},
#> pages = {299--314},
#> year = {1996},
#> url = {https://doi.org/10.1080/10618600.1996.10474713}
#> }
#> @book{R,
#> author = {R {Core Team}},
#> title = {{R: A Language and Environment for Statistical Computing}},
#> publisher = {R Foundation for Statistical Computing Vienna Austria    :},
#> year = {2016},
#> url = {https://www.R-project.org/},
#> isbn = {3-900051-07-0}
#> }
```

Authors who are converting a bibliography from ‘bbl’ should check if there are any errors or missing values.

3 General usage

General use of the **rebib** package is quite easy, requiring only one function call.

Using with RJournal article

In bibliography construction mode, **rebib** will check for existing BibTeX files and will convert the embedded bibliography only when there are no linked BibTeX files in the RJ article.

```
%% typically rebib will search for this line in the RJ article,
%% not the RJwrapper file
\bibliography{example}
```

If there is no linked BibTeX file, **rebib** will begin its procedure of converting the embedded bibliography to BibTeX and then link it with the article file as well.

```
# for files without BibTeX source
rebib::handle_bibliography(your_article_path)
#> Warning in get_bib_file(article_dir, file_name): No Bib files found !
#> BibTeX file does not exist
#> will parse for bibliography
#> bibtex file created
cat(readLines(paste(your_article_path,"example.bib",sep="/")),sep = "\n")
#> @book{ihaka:1996,
#> author = {{Ihaka, Ross and Gentleman, Robert}},
#> title = {{R: A Language for Data Analysis and Graphics}},
#> publisher = {Journal of Computational and Graphical Statistics 3:   :},
#> pages = {299--314},
#> year = {1996},
#> url = {https://doi.org/10.1080/10618600.1996.10474713}
#> }
#> @book{R,
#> author = {R {Core Team}},
#> title = {{R: A Language and Environment for Statistical Computing}},
#> publisher = {R Foundation for Statistical Computing Vienna Austria    :},
#> year = {2016},
#> url = {https://www.R-project.org/},
#> isbn = {3-900051-07-0}
#> }
```

Bibliography aggregation

If you have a BibTeX file and it is missing some references then this mode can help you read the embedded bibliography and combine the two bibliographies in a single BibTeX file.

```
# for files with BibTeX source as well as embedded entries
rebib::aggregate_bibliography(your_article_path)
#> Found Bib file  example.bib
#> bibliography aggregation possible
#> Found Bib file  example.bib
#> aggregation delta : 2
#> BibTeX file aggregated
```

Using with any LaTeX/.bbl file

The bibliography converter will convert any LaTeX or '.bbl' file given as a path. This way, one can generate BibTeX from the embedded bibliography or '.bbl' file.

```
rebib::biblio_converter(path_to_your_file)
#> working directory : /temp/Rtmpt6Qk0v/exampledir/standalone
#> bib entries : 6
#> Written BibTeX file : /temp/Rtmpt6Qk0v/exampledir/standalone/sample.bib
```

4 Limitations

The limitations of **rebib** package are:

- The **rebib** package only supports embedded bibliography format suggested/ commonly used in R Journal.
- Encoding and character set issues might not work with rebib well. Try to use UTF-8 encoding if possible.
- Bibliography aggregation won't fix existing issues like missing commas/brackets.

5 Summary

In summary the **rebib** package supports:

- Conversion of embedded natbib type bibliography entries used in the R Journal
- Aggregation of bibliographic entries from the article as well as separate BibTeX files

Abhishek Ulayil
Student, Institute of Actuaries of India
Mumbai, India
ORCiD: 0009-0000-6935-8690