User guide for the `dumbib` package*

Shivam Garg

July 11, 2024

The `dumbib` package helps with bibliography management for LaTeX, by creating forward links (when you cite a paper, the citation links to the bibliography entry in the reference section) and backward links (the bibliography entries link back to all the places in the document where they were cited). **This is the only functionality that this package provides**; to remind users of how minimalistic `dumbib` is in its processing, it even has the word "dumb" (or "dum" for dummy, depending on how you view it) in its name.

Owing to this minimalistic processing, `dumbib` gives its users a near complete control over how their bibliography looks like, which is its primary appeal. This is in contrast to popular bibliography packages such as BibTeX or BibLaTeX, which provide a lot of functionality but require additional effort from the users to tweak the program output to their exact liking. Another idea behind `dumbib` is that a lot of this manual work, in particular the creation of a `dumbib` database (see §1), can be automated using, say, a `Python` script. (The nice thing about this division of labor is that programming in `Python` should be more desirable to most users than tweaking other bibliography packages or, worse, programming in LaTeX.)

To use `dumbib`, ensure that the `dumbib.sty` file (which is available at the link given in the footnote at the bottom of this page) is in the path of your system and then include the package in the preamble of the TeX document as follows:

```
\usepackage{dumbib}
```

The package provides three commands for bibliography management:

1. `\dumbibReferenceEntry{}` for creating the bibliography database,
2. `\cite{}` for citing the references in the text, and
3. `\dumbibCreateBibliography`: for laying out the references in a reference section.

*The package is available at the following link: https://github.com/svmgrg/bibtex_alternative.

1 Creating a bibliography database

At the beginning of your document you will need to create a database of all the references you plan to use in your document so that `dumbib` is aware of them. This is done using the command

```
\dumbibReferenceEntry{<key>}{<author>}{<year>}{<citation_text>}
```

which has four mandatory arguments:

- `<key>` is the citation key which you will invoke when you want to cite this reference,
- `<author>` and `<year>` refer to the author(s) and year that appear in-text, when you cite this reference, and
- `<citation_text>` is the bibliography entry that will go in the reference section.

The following snippet adds two references to the `dumbib` database in an APA like style:

```
\dumbibReferenceEntry{lowry_etal1951}{Lowry et al.}{1951}{
  Lowry, O. H., Rosebrough, N. J., Farr, A. L., Randall, R. J. (1951).
  Protein measurement with the Folin phenol reagent.
  \textit{Journal of Biological Chemistry, 193}(1), 265-275.
}

\dumbibReferenceEntry{noorden_etal2014}{Van Noorden, Maher, and Nuzzo}
{2014}{
  Van Noorden, R., Maher, B., Nuzzo, R. (2014).
  The top 100 papers. \textit{Nature News, 514}(7524), 550.
}
```

This snippet goes **after** the `\begin{document}` command in your TEX file. (If you prefer, you can instead put this `dumbib` database in a separate file, say, `database.tex`, and include that in your main TEX file using the command `\include{database.tex}`.)

Note that you are free to specify exactly how the citation appears in-text and in the reference section. For instance, you could have instead modified the last citation above to the following:

```
\dumbibReferenceEntry{noorden_etal2014}{Van Noorden et al.}{2014}{
  R. van Noorden, B. Maher, R. Nuzzo. (2014).
  The top 100 papers. \textit{Nature News}.
}
```

However, since you have complete freedom in deciding exactly how the citation appears in-text, you have to manually take care of things which other bibliography systems would have automated for you. For instance, if you are including multiple references from the same author(s) in a single year, you will **manually** need to append `'a'`, `'b'`, etc. to the year of publication for these references:

```
\dumbibReferenceEntry{bach2023a}{Bach}{2023a}{
  Bach F. (2023a).
  \textit{Learning theory from first principles.}
  MIT press.
}

\dumbibReferenceEntry{bach2023b}{Bach}{2023b}{
  Bach F. (2023b).
  On the relationship between multivariate splines and infinitely-wide
  neural networks.
  \textit{arXiv:2302.03459.}
}
```

(Even though in the above examples, the author list, the paper title, and the publication venue appear in separate lines inside `\dumbibReferenceEntry{}`, this was done only to improve readability; `dumbib` doesn't care if they all were formatted in a single line.)

2 Error handling

Whenever `dumbib` encounters an error, it prints a succinct warning message on the terminal, and a verbose error message that will **ostentatiously appear in the PDF document**. For instance, if you create two different reference entries with the same key using the `\dumbibReferenceEntry{}` command, then `dumbib` will raise an error and **ignore the second entry all together**, i.e. the citation key will continue pointing to the reference that was declared first. This behavior is shown below. (Although, `dumbib` will gladly accept two identical references as long as they have a different key; it is really dumb!)

The following snippet uses the key "`talagrand2022`" twice resulting in an error.

```
\dumbibReferenceEntry{talagrand2022}{Talagrand}{2022}{
  Talagrand, M. (2022).
  Upper and lower bounds for stochastic processes: Decomposition
  theorems. \textit{Springer Nature.}
}
```

```
\dumbibReferenceEntry{talagrand2022}{Empty}{0000}{
  Empty example reference.
}
```

This is the succinct warning message that is printed in the terminal:

```
Package dumbib Warning: Error on line 161! Key `talagrand2022' already
(dumbib)                defined.
```

This is the verbose error message that will appear in your output PDF exactly as follows:

Error on line 161! The key "talagrand2022" has been already defined. Consider using a different key name.

Original entry: Talagrand, M. (2022). Upper and lower bounds for stochastic processes: Decomposition theorems. *Springer Nature.*

New (ignored) entry: Empty example reference.

The above error message is, by design, excessively intrusive so as to attract the user's attention to the errors. Once the user solves the corresponding error, the message will disappear.

Raising errors only on the terminal

In case you do not want to see such verbose messages in the output PDF document, you can load the `dumbib` package as follows:

```
\usepackage[error_mode]{dumbib}
```

In this mode, `dumbib` will not show any errors in the output PDF, and instead raise the succinct message as an error on the terminal, i.e. it will break the LaTeX processing and require the user to type `<return>` to continue.

3 Citing references in-text

The command to cite the references in-text is

```
\cite[*][<optional 'a' or 'y'>]{<key>}
```

This command takes the citation key, along with an optional star, or an optional `'a'` or `'y'` argument if you want to just print the author(s) or the year of publication. The `\cite{<key>}` command will appear in-text as `'<author> (<year>)'`, where the `<author>` and `<year>` fields correspond to those that were specified during the `dumbib` database creation. And the starred version `\cite*{<key>}` produces the output `'<author>, <year>'`.

For instance, the snippet

```
According to \cite{noorden_etal2014}, the most cited paper in recorded
history is a biology paper by \cite[a]{lowry_etal1951}\ that has been
cited hundreds of thousands of times. Unfortunately, we don't conduct
experiments involving proteins, and are thus more likely to cite some
probability or machine learning texts (such as \cite*{talagrand2022};
\cite*{bach2023a}; \cite[y]{bach2023b}). In particular, the book by
\cite{talagrand2022} seems quite nice and surprisingly readable.
```

produces the following text:

> According to Van Noorden, Maher, and Nuzzo (2014), the most cited paper in recorded history is a biology paper by Lowry et al. that has been cited hundreds of thousands of times. Unfortunately, we don't conduct experiments involving proteins, and are thus more likely to cite some probability or machine learning texts (such as Talagrand, 2022; Bach, 2023a; 2023b). In particular, the book by Talagrand (2022) seems quite nice and surprisingly readable.

As the above example shows, **the user needs to manually take care of everything down to the very basic details**. For instance, in the above example, we provided the opening and closing parentheses, the semi-colons, and even ensured that only the year of publication appeared in the last citation (Bach, 2023b). We also had to put a backslash after the command `\cite[a]{lowry_etal1951}` to ensure optimal spacing following a period, just like we would have done had we typed the entire thing manually, i.e. `Lowry et al.\`. If there are any errors in the usage of the `\cite{}` command (such as when the user provides an incorrect optional argument, or tries to cite a reference which had not been declared to the `dumbib` database a priori) they will be raised appropriately, as was previously discussed.

Note: To create the links, `dumbib` uses the `hyperref` package under the hood. By default, such links would appear as colored bounding boxes. Instead, if you want your links to appear as dark red-colored text without any bounding boxes (as shown in the example above), you can include the following in the preamble of your TEX file:

```
\hypersetup{
  colorlinks,
  linkcolor={red!33!black}
}
```

4 Laying out the bibliography section

The following command lays out the references in the bibliography section:

```
\dumbibCreateBibliography[*]
```

The starred version of this command does not produce backward links to the places where the papers were cited in-text.

The following snippet can be used to create a reference section similar to the one that appears at the end of this page.

```
\section*{References} % \addcontentsline{toc}{chapter}{References}
\dumbibCreateBibliography
```

The `\dumbibCreateBibliography` command raises an error if a bibliography entry were created in the database but never cited in-text. Also, as can be seen below, **the references are displayed in the same order in which they were declared using the `\dumbibReferenceEntry{}` command** during the creation of the `dumbib` database. Therefore, if you want the references to appear in an alphabetical order, you should declare them alphabetically. Also note that the style of references below is not consistent, because during their declaration, we didn't follow a consistent style. Further, if a reference is cited multiple times (such as the book by Talagrand, 2022) on the same page, it will appear multiple times in the backward links as well (see the last reference below).

References

Lowry, O. H., Rosebrough, N. J., Farr, A. L., Randall, R. J. (1951). Protein measurement with the Folin phenol reagent. *Journal of Biological Chemistry, 193*(1), 265-275. (cited on page 5)

Van Noorden, R., Maher, B., Nuzzo, R. (2014). The top 100 papers. *Nature News, 514*(7524), 550. (cited on page 5)

Bach F. (2023a). *Learning theory from first principles.* MIT press. (cited on page 5)

Bach F. (2023b). On the relationship between multivariate splines and infinitely-wide neural networks. *arXiv:2302.03459.* (cited on pages 5 and 5)

Talagrand, M. (2022). Upper and lower bounds for stochastic processes: Decomposition theorems. *Springer Nature.* (cited on pages 5, 5, and 6)