Academic Writing in LaTeX: Best and Worst Practices

YEGOR BUGAYENKO

This is a humble attempt to summarize most typical mistakes we make while writing academic papers in LaTeX and most important recommendations. Each suggestion or a mistake takes a short paragraph of description right here and also may suggest looking into a more detailed explanation in some other online resource. We recommend, before submitting your paper to a conference or a journal, go through this list of mistakes and make sure none of them are present in your paper.

LaTeX sources of this document you can find in this GitHub repository and contribute your ideas through a pull request.

Beforehand, we suggest you read these:

- Technical Writing Courses by Google

§1: Check your `.tex` sources with lacheck and maybe other tools.

1 PREAMBLE

§2: Use `acmart` document style and read their Best Practices. Start the document with this:

```
\documentclass[11pt,nonacm,natbib=false]{acmart}
\settopmatter{printfolios=false,printccs=false,printacmref=false}
\usepackage[maxnames=1,minnames=1,natbib=true,
  citestyle=authoryear,bibstyle=authoryear]{biblatex}
\addbibresource{main.bib}
```

§3: Use `biblatex` and `biber`, here is why. Place your citations into `main.bib` file. Later in the document print the bibliography with `\printbibliography` command.

2 HEADINGS

§4: Capitalize all nouns and verbs in headings, here is why and how.

3 TYPOGRAPHY

§5: Use single dash inside words, e.g.: `micro-service`. Use double "endash" between numbers, e.g.: `pages 15--28`. Use triple "emdash" between words avoiding spaces, e.g.: `We---since you ask---disagree`. Read this.

4 FONTS

§6: Prefer `\emph` over `\textit`, here is why.

§7: Avoid `\textbf` and all other font changing commands at all cost. Here is my rant on this very problem of technical people trying to make their products look visually attractive and failing miserably.

5 COLORS

§8: Don't use them. Keep your documents strictly black-on-white. Read more about this.

6 CODE SNIPPETS

§9: Use `ffcode` package, which allows writing both code snippets and fixed-width-font in-paragraph text blocks.

7 FIGURES AND TABLES

§10: Don't force positioning in figures and tables, like `\begin{table}[h]`. Instead, just `\begin{table}`.

§11: Make sure the explanation you place into `\caption` is detailed enough to let your reader understand the content without searching the text; see how it's done in this paper.

§12: Prefer a list over a table and a table over a graph.

§13: Align text cells by left, center headings, and align cells with numbers by right (read this discussion); here is a more detailed discussion. Here is an example of a table properly formatted:

Name	Age	Role
Jeff	35	The creator of the main algorithm and the owner of the code
Sarah	38	The architect of all microservices and the developer of Java modules

```
\documentclass{article}
\usepackage[paperwidth=3in]{geometry}
\pagestyle{empty}
\usepackage{booktabs}
\usepackage{tabularx}
\begin{document}
\begin{tabularx}{\columnwidth}
  {lr>{\raggedright\arraybackslash}X}
\toprule
Name & Age & Role \\
\midrule
Jeff & 35 & The creator of the main
algorithm and the owner of the code \\
Sarah & 38 & The architect of all
microservices and the developer of
Java modules \\
\bottomrule
\end{tabularx}
\end{document}
```

§14: Put all tables into `table` environment:

```
\begin{table}
.. content goes here
\caption{Caption}
\label{tab:my-table}
\end{table}
```

8 BULLETS

§15: Prefer in-paragraph itemization over a vertical one and use `paralist`:

The following sources were analyzed: 1) GitHub, 2) Google, and 3) Stack Overflow.

```
\documentclass{article}
\usepackage[paperwidth=3in]{geometry}
\pagestyle{empty}
\usepackage{paralist}
\begin{document}
The following sources were analyzed:
\begin{inparaenum}[1)]
\item GitHub,
\item Google,
and
\item Stack Overflow.
\end{inparaenum}
\end{document}
```

§16: In all itemization use Oxford comma, as in the list above before the "and" (provided there are more than two items).

9 URLS

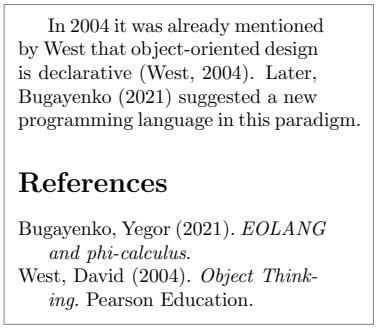
§17: Use `href-ul` package and then `\href` command.

10 REFERENCES

§18: Don't use `\ref`, use `\cref` instead from `cleveref` package.

11 CITATIONS

§19: This code demonstrates how to use APA-style citations with `natbib` commands:



In 2004 it was already mentioned by West that object-oriented design is declarative (West, 2004). Later, Bugayenko (2021) suggested a new programming language in this paradigm.

References

Bugayenko, Yegor (2021). *EOLANG and phi-calculus.*
West, David (2004). *Object Thinking.* Pearson Education.

```
\documentclass{article}
\usepackage[paperwidth=3in]{geometry}
\pagestyle{empty}
\usepackage[natbib=true,citestyle=authoryear,
  bibstyle=authoryear]{biblatex}
\addbibresource{main.bib}
\begin{document}
In \citeyear{west2004} it was already
mentioned by \citeauthor{west2004} that
object-oriented design is
declarative~\citep{west2004}. Later,
\citet{eolang2021} suggested a new
programming language in this paradigm.
\printbibliography
\end{document}
```

§20: Place `~` (tilde) symbol before `\citep` to avoid line breaks, see why.

§21: Prefer `\citet` over `\citep`, making references more obvious, as in the second sentence in the example above.

§22: Don't type author names or reference titles directly, only use `\cite*` commands.

§23: Remember that brackets are not words.

§24: Don't cite Web pages or any other URLs.

§25: Add bibcop to your document, to make sure `.bib` file is properly formatted.

12 REFERENCES

§26: The references in `.bib` file are usually imported from Google Scholar or similar sources. Unfortunately, such imports often contain typos and mistakes. Check the items printed in the "References" section for the following:

- Year is not printed twice;
- Dashes in titles are printed as `---` without surrounding spaces;
- All nouns and verbs are capitalized in all titles.

§27: Use biblint to check your `.bib` file.