How to prepare academic documents avoiding Word

Gerardo Marx Chávez-Campos

February 2020

Introduction

The preparation of documents always takes our minds to relate the usage of specialized software like Microsoft-Word, Word-Pad, Pages, Open-office, and so on. However, many years ago, the preparation of documents, used to be more relaxed and faster; the plain-text fashion. Plain-text documents are files with no format nor specialized information. Instead, plain documents are only text. But, why plain-text documents are so crucial for the academic area? Why these files remain useful? And how can we use them to create high-quality documents? Well, let's start by explaining some special kinds of documents based in plain-text files; the *Markdown files*.

Formats for academic documents

The Markdown format was initially developed by John Gruber in collaboration with Aaron Swartz, to simplify the writing of HTML. Instead of coding a file in HTML syntax, the content of a document is written in **plain text** and denote with *simple tags the final formatting*. Subsequently, the MD (Markdown) files are parsed to generate the final HTML document. With this concept, the source file remains easily readable, and the **author can focus on the contents rather than formatting**.

Why MD files are important in academia?

Despite its original focus on the web, the MD format has been proven to be great for academic writing. In particular, pandoc-flavored MD adds several extensions that facilitate the authoring of academic documents and their conversion into multiple output formats. The only requirement is to learn the basics of MD-syntax and install the *pandoc* application to convert the files. Then, the output file can be defined by using some parameters in the input file to create a new different output document. Therefore, it is possible to create tables, bibliography citations, include figures, code sections, special symbols and characters, and the most important only using a plain-text file.

Why these files remain useful now?

A single word can answer this question: Reproducibility. An MD file can be compiled..., yes compiled; we are talking about programming, ... compiled in a different computer, if the pandoc engine is installed on it. And guess what? We can add one more word: traceability. Yes, we can follow how a file is changing its content and support old and new versions. -Hey, check my paper please, this is only a simple plain text, and you will not have problems of compatibility-, right? In most of the cases, there should no be a problem

How can we use them to create high-quality documents?

By using pandoc there is possible to produce '.tex' files and then produce a PDF-formated document. When the TeX file is generated, the document's quality can be improved. Even more, the TeX file can be basically modified to create an entirely different document by selecting a new 'document class'. Nevertheless, from my point of view, the MD files can be used for basic and short documents (proposal, drafts, guides, unformatted articles). In contrast, for more complex documents (thesis, books, formated-articles), it is better to use LaTeX.

Whats next?

Pandoc is useful for creating basic and short documents; also is a first approach to use more powerful engines like pdflatex and xelatex. Also, the MD files are just plain text files and can be used for controlling versions. Actually, the MD file's process is commonly used by other IDE systems like r-studio to create complete environments for data science reproducibility.