Your Dissertation Title

Your Full Name

A DISSERTATION

Presented to
The Department of Biomedical Engineering
Oregon Health & Science University
School of Medicine

In partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

August, 2020

Certificate of Approval

To someone important

Contents

List of Figures

List of Tables

Acknowledgements

Write a nice sappy letter to the important people in your educational experience.

Abstract

Why use LaTeX? Great writing deserves to be presented beautifully, and whether you think it fair or not, first impressions and aesthetics do matter when presenting important work. Text should be evenly spaced and correctly wrapped around interjecting figures and tables, mathematical equations must be clean and consistent, and references should be so simple to use that they don't distract from the writing process. The vast majority of modern publishers type-set their works in LaTeX, and for good reason: the system is designed from the ground up to make writing look visually appealing, consistent, and precise. Though Microsoft Word is a fine tool for general use document preparation, LaTeXis for science and publishing. Even if you, the author of a scientific paper, have submitted a document to a publisher in a Word format, that publisher type-sets your writing prior to printing. In fact, for many reasons, the LaTeX document engine has been likened to the Gutenburg Printing Press as an essential tool of the modern publisher.[1]

The obvious and most important distinction between a document type-set with LaTeX and a general text renderer like Word is that LaTeX compiles documents in a manner similar to the way a compiler builds software programs: it optimizes the document at every step to make sure lines are even, text is distributed, and formatting is obeyed uncompromisingly. For example, notice how consistent the block-justified text is in this document. While Microsoft Word's block justification algorithms format each line independently, rendering horrendously inconsistent spacing between words within the same paragraph, LaTeX solves this problem by optimally spacing not just words on a single line, but every element in the entire document to generate a consistent look and feel. Magic!

This document is intended to serve as a skeleton for a PhD dissertation presented to Oregon

[1] `http://www.practicallyefficient.com/2017/10/13/from-boiling-lead-and-black-art.html`

Health & Science University, though the basic structure can easily be adapted to different institutions. Although intimidating at first, the vast majority of regular LaTeX usage is simple and intuitive. Although LaTeX makes liberal use of style-files typically designed by the publisher, this example uses a basic style template included in the LaTeX software, so significant modification should not be necessary.

Here are a few helpful points to keep in mind. When writing LaTeX documents, write one sentence on each line in the `.tex` files; don't worry, they wrap around in the editor. Similarly, Paragraph breaks are designated by skipping a line. Use an integrated editor/viewer, such as TexMaker (free and open-source, of course). References and bibliographies effectively take care of themselves, but you need to use a reliable citation manager. Mendeley is a great, free choice. Organize the skeleton of your document in `main.tex`, and keep separate directories for each chapter and figures. Staying organized can make life even better!

On a personal note, I forced myself to learn LaTeX syntax after I turned in my Master's thesis several years ago to another institution. At that time, I was fed up with Word's drag-and-drop formatting implosions, unreliable equation editor, and messy citation manager, so my adviser mercifully introduced me to the LaTeX system; my life's relationships with writing documents has been much better ever since. I admit I am something of a LaTeX evangelical fan-boy, and secretly harbor a sense of superiority whenever I glance at someone else's important document and know immediately that it was prepared in Word based solely on how unpleasant it looks; regrettably, I make no apologies for myself. Whether you harbor a sense of superiority for your document or not, I wish you confidence in your published material made possible through this tool.

This document is a minimal example with simple illustrations of how to embed images, tables, and references into your own document. If you want to do something fancy, or you require a more detailed explanations of how LaTeX works under the hood, consult your nearest Google search bar. The contents herein are provided free of charge, but as always, you get what you pay for.

Happy type-setting!

Geoffrey F. Schau

May 2020

Chapter 1

Introduction

> Everything that can be invented has been invented.
>
> *Charles H. Duell*
> *Commissioner,*
> *U.S. Office of Patents (1899)*

To get started, it's probably best that you download and install TexMaker. Within TexMaker, you should open `main.tex` as the file that you want to actually compile. In that file you can edit anything you want, but remember that each chapter explicitly points to its own document to keep your `main.text` nice and clean. This chapter walks through the basics.

Preceding text can preface section headers. Observe how spacing is consistent, even in difficult situations that would otherwise require an inconsiderate line or section break. By compiling the entire document, the sophisticated algorithms under the LaTeX hood have automatically rendered a consistent document.

Remember to keep your `.tex` files as well-organized as you can. Each line in the file should correspond to one sentence. Paragraph breaks are designated by skipped lines, and full line breaks in text are designated with two back-slashes like this.

See, a full line break.

1.1 A basic section header

Text goes here. If you don't like the epigraph above, replace it, or delete it. By default, new sections are numbered. Don't like numbers? Use an asterisk to remove them, like this:

Unnumbered section header

This is now a new section. Importantly, unnumbered sections are not included in the table of contents. By the way, do you see how easy it is to make the table of contents, figures, and tables? A single line in LaTeX takes care of innumerable minutes of frustration in Word.

Need subsections? Easy beans.

1.1.1 Here's a subsection

Sub-subsections go deeper still

Although common convention is to halt display of sub-indices in the section header, this can be changed. I don't remember how to, but Google can explain. It's not hard, I promise.

Intermediate text is ignored in the table of contents, but the section and subsection headers are automatically included

1.2 Text Formatting

You can make words **bold** or *italicized* very easily and explicitly. As is almost always the case, LaTeX supports nested operations, so ***anything*** can be modified however you want. I like to use `mono` text for things like variable or URLs.

1.3 Figures

To include a figure in your document, you must first save the figure in a good location to which you can point LaTeX. A figure is first defined and then populated with the graphics object, a caption, and a label. The label is important, as it allows you to effortless refer to the specific figure from anywhere in the text withour needing to keep track of anything yourself, by referencing Figure 1.1

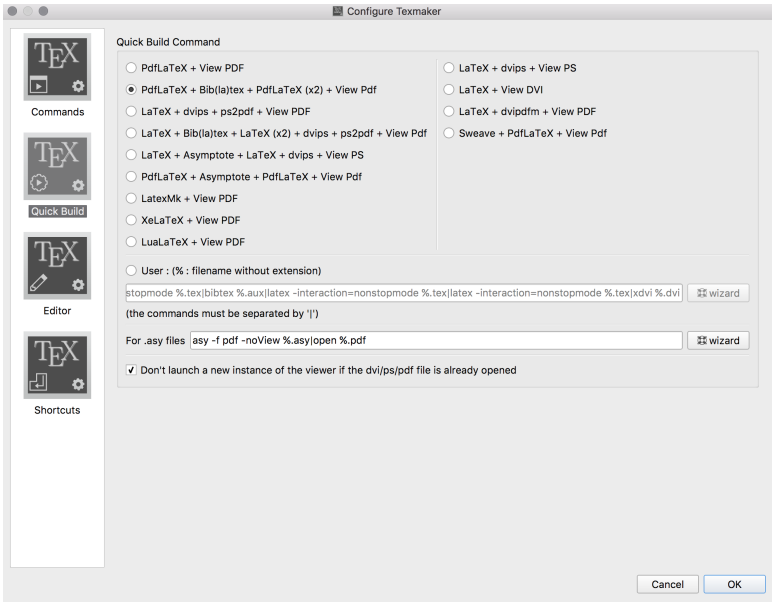

Figure 1.1: Example figure - TexMaker configuration option

just like that. You can control the placemenet of figures on the page by placing variables in the square brackets following the `\begin{figure}` line, such as `\begin{figure}[hbt]`, where `h,b,t` designate "here", "bottom", and "top", respectively. Note that in `main.tex`, I explicitly designate a graphics path, so the pointer in `includegraphics` should only require the figure name, not the full path. Easy!

Pointing a document to a file on a computer when compiling is far superior to dropping the file into a document for many reasons, but risks abound. On one hand, this approach is elegant, because if you execute code that generates figures, you can keep the pointer the same in the LaTeX document so that each time it is compiled it will automatically load the most recent figures on your filesystem. However, if you ever remove figures from the path, then the LaTeX pointers get confused. In my experience, it's a good idea to copy figures from your disk into the latex document directory rather than linking them.

1.4 Tables

Tables can become complicated quickly, but a few key properties make tables more readable than others. The table should be top and bottom-ruled with a thicker boarder while intermediate line breaks should be used sparingly, and done with a thinner boarder. Text should not bleed into the

margins, and should be rotated onto their own page if necessary. Just like figures, they can be referenced in the same way, like referencing Table 1.1.

1.5 Equations

Equations can obviously get very complicated, but the basic structure is to wrap your equation in a similar manner to figures and tables with a reference label, so you can reference things like Equation 1.1 or 1.2.

$$a = 10 \tag{1.1}$$

Here's an example of the pretty equation describing the loss term of my favorite deep learning model.

$$\mathcal{L}_i(x_i, \theta, \phi) = -\mathbb{E}_{z \sim q_\theta(z|x_i)}[\log p_\phi(x_i|z)] + \mathrm{KL}(q_\theta(z|x_i) \| p(z)) \tag{1.2}$$

See Google for any specific help writing equations. The syntax can feel like a lot, but the syntax for generating complex equations is extremely powerful and is the gold standard for mathematical writing.

1.6 Lists

LaTeX has a myriad of mechanisms to generate lists, but the two most common are the bullet-style list and enumerated lists.

- Itemized lists and enumerated lists are identical except for the element's header
- Simply replace `itemized` with `enumerated`

Table 1.1: An example table

	VAE				
	PC1	PC2	LD1	HCF	Cell Count
F value	717.9	8.2	1073	254	431.5
Pr (> F)	< 2e-16*	2.83e-4*	< 2e-16*	< 2e-16*	< 2e-16*

But, as is always good practice, multiple instances of tables, figures, or lists should be separated by some sort of text.

1. See, easy as that.

2. In both cases, sub-bullets are created simply by defining nested itemized or bullet lists

    - In general, the leading characters are automatically updated for you, but can be modified any way you wish.

        (a) See Google for more examples

1.7 Citations

The bibliography system in LaTeX is one of its greatest features by making life immesurably easier than my previous experience with Word. To generate a bibliography, select the relevant documents in Mendeley or Zotero and export the selection as a `.bib` file format, which includes all necessary metadata about the citation plus a unique handle to refer to within a LaTeX document. I have included a `bib.bib` file from a random selection of documents in the root on this project. You can cite as many or as few as you want in the text, and LaTeX will pull them automatically and collate them into a bibliography at the end of this document, with citations made as simple as this.[1]

See the `main.tex` file to see how easy it is to point LaTeX to your bibliography. Everything else works like magic.

1.7.1 Footnotes

I like footnotes. I think they are fun to read and write, and give voice to the tangential thoughts of an otherwise omnipresent author. Footnotes are easy in LaTeX, and can be used for quick references or to make a parenthetical point.[1] If you are a fan of judicial legal opinions or the writing of David Foster Wallace (let's be honest, we all are), then nested footnotes might be your bag.[2]

[1] I don't have a reference, but look how easy footnotes are in LaTeX!

[2] But honestly, they're probably not a good idea for scientific writing.[3]

[3] Unless you really want to lead your reader through a long parenthetical maze of free-thinking, minimize the use of footnotes in a dissertation.[4]

[4] In fact, it's probably better that you don't use them at all

1.8 Comments

Remember that just like computer code, you can comment out text in latex by appending a % character in front of the line. This is also why keeping one sentence per line makes sense. A quick note on special characters in-text: because LaTeX reserves certain characters, like the percent sign, for special use, most special characters require a preceding backslash. So things like % and & need to be specifically designated. Google can provide a full list of special characters.

1.9 Epigraphs

I have included epigraphs for each chapter (those little quotes at the beginning). I like them, some don't, remove them if you desire. Some people also don't like the Oxford comma; those people do happen to be wrong and misguided.

1.10 Misspellings

Here is where I sheepishly tuck my evangelical tail. I admit that LaTeX does not have great support for identifying spelling or grammar mistakes. In a pure "what you see is what you get" philosophy, LaTeX allows you to do whatever you want without imposing itself on how you should be doing them. This means it stays out of the way when you make a spelling mistake. The solution that I find most simple is to just copy the text in the `tex` file and paste it into Word so you can go through and quickly find the little red squiggly lines.

1.11 Errata

If there is anything in this document that you would like to see modified, please let myself or the current document owner know. This may be hosted on github at some point for anyone to copy and easily get started with, but I will save that for a later day.

If you found this helpful, tips accepted at the following Bitcoin wallet address:

`1KTY4tQYbsdXYHNj9kHpe8UKMDEnncAaBw`

Chapter 2

Chapter Two Title

An illuminating quote.

Your favorite philosopher

2.1 Introduction

2.2 Methods

2.3 Data

2.4 Results

2.5 Discussion

2.6 Conclusions

Chapter 3

Chapter Three Title

An illuminating quote.

Your favorite philosopher

3.1 Introduction

3.2 Methods

3.3 Data

3.4 Results

3.5 Discussion

3.6 Conclusions

Chapter 4

Chapter Four Title

An illuminating quote.

Your favorite philosopher

4.1 Introduction

4.2 Methods

4.3 Data

4.4 Results

4.5 Discussion

4.6 Conclusions

Chapter 5

Conclusion

But... what is it good for?

Advanced Computing Systems Engineer,
commenting on the microchip
IBM (1968)

In conclusion, use LaTeX or don't.

Bibliography

[1] Chun Han Lin, Tiina Jokela, Joe Gray, and Mark A. LaBarge. Combinatorial Microenvironments Impose a Continuum of Cellular Responses to a Single Pathway-Targeted Anti-cancer Compound. *Cell Reports*, 21(2):533–545, 2017.