How to convert a latex document into a Word .docx
Author: Laura Tremblay-Boyer, l.boyer@fisheries.ubc.ca

Note 1: This was written for Mac/UNIX users, but most instructions should be valid for Windows.

Note 2: There are many ways to do this, and I have found none at this stage that are perfect. The best option in my opinion is to use latex2rtf (see section 1) which converts .tex to a .rtf (Rich Text File), which is in turn readable by Word. The manual changes that you have to do for the latter are minimal, but it is a tad confusing to install. Once that hurdle is past though, you are good to go. I included instructions for another option, Pandoc (see section 2), which is very easy to install and can handle conversions between many, many file formats – but the resulting .docx lacks some features that are a pain to correct manually (e.g. in-text references). Still, it is a good ressource to have given its versatility.

Both latex2rtf and Pandoc requires you to use the command line, via the Terminal (which apparently has an equivalent in Windows called PowerShell? ...unverified). To find it, type "Terminal" in the Spotlight search bar.

→ to view the instructions for *latex2rtf*, click here.
→ to view the instructions for *Pandoc*, click here.

1   The almost-quick-and-easy-and-almost-perfect way: latex2rtf

I wrote this for Mac/UNIX users, see this webpage for instructions for Windows: http://latex2rtf.sourceforge.net/usage.html.

1.1   Installation

1. Download latex2rtf and unpack it: http://sourceforge.net/projects/latex2rtf/

2. Open the terminal and navigate to the unpacked `latex2rtf` directory, for e.g.:
   `cd  /Downloads/latex2rtf-2.3.3/`

3. Once in the directory, type the following commands one at a time:

   `make`

   `make check`

   `make install`

   If you get a 'Permission denied...' message here, try adding *sudo* before the command (it will ask for the administrator's password, and should then proceed normally):
   `sudo make install`

4. Verify that latex2rtf installed properly by typing:
   `latex2rtf -v`
   ... if this produces something intelligible, you are good to go.

5. If this didn't work for you, darn. You probably have to make changes to the Makefile. Open in a text-editor the README that comes with the program for further instructions. If you are using a Mac you might get a message that this comes from an unidentified developer and cannot be opened. CTRL-click Open the file name and it should let you open the file despite the warning.

1.2 Converting .tex to .RTFs

1. Open the Terminal and go to the directory containing the latex file you want to convert:
   `cd /Documents/.../my-directory-of-choice`

2. (Optional) Once in the directory, you can use `ls` to list the files it contains — you should be able to see the file you want to convert.

3. Create a copy of your file (just in case, but probably not necessary).

4. Type the following command:
   `latex2rtf your-file-name.tex`
   ... this should produce the file `your-file-name.rtf`. Note the .rtf extension. During the conversion, watch out for warnings of the type "Package/option ... unknown" – you will have to modify manually whatever settings were changed by these options as they were not imported in the .rtf.

5. Open the `your-file-name.rtf` in Word... Most things should be there, but have a scan through to see if anything is missing. For instance if you used the package `fullpage` you will have to change the margins of the Word document back to 2.5cm. If you had pages in landscape format you'll have to add those manually by adding a section break before the desired landscape page and changing the orientation for that page only.

6. When you are done use Word to save the .rtf to a .docx. And you are done. Party!

2 The kinda-quick-and-easy-but-not-perfect way: Pandoc

This method handles 95% of formatting, figures and bibliography — and it comes with its own installer (just double-click the .dmg). It doesn't do mixed portrait and landscape pages, in-text references to figures and tables, and probably other stuff that I can't think of right now. However it can convert between a number of file formats, see http://johnmacfarlane.net/pandoc/diagram.png (... *right?!*).

Once you have specified the right options, the output .docx doesn't require manual alterations. It is probably good for sending to a supervisor or collaborator (who you will have forewarned of minor formatting issues) but not as the final version to, say, a thesis committee or a journal.

Here goes:

1. Download and install Pandoc: http://johnmacfarlane.net/pandoc/installing.html

2. Open the Terminal and confirm that Pandoc is installed by typing:
   `pandoc --version`
   (if it is not installed it will print `command not found`).

3. Navigate to the directory containing the file you want to convert using the command `cd` (use `pwd` to see where you are presently):
   `cd /Documents/Manuscripts/`

4. (Optional) Once in the right directory, you can use `ls` to list the files in the directory — you should be able to see the file you want to convert.

5. Create a copy of your file (just in case).

6. Now, tell pandoc to convert your file from latex to docx:
   `pandoc your-file-name.tex -f latex -t docx -s -o output-file-name-of-your-choice.docx`
   `<other options as needed >`

   - -f <...>: the format you are converting *from*

- -t <...>: the format you are converting *to*
- -o <...>: the desired *output* file name, with the extension

7. To see the many options that can be added to the pandoc command, go here: http://johnmacfarlane.net/pandoc/README.html#options.

8. Some miscellaneous things that should come in handy:

   - Bibliography: If you have a bibliography, tell Pandoc where to find it by adding the option `--bibliography = your-bibliography-file-name.bib`.

   - Figures: At this stage your document looks pretty good *except* for wonky figure sizes which do not match those you defined in your .tex (shoot). Here's why: Pandoc imports the figure at its original size and ignores the size settings you defined in the .tex. Thanks Pandoc. The easy way around this (and this is only a pain once) is to make sure that you always produce your figures in a size that is adequate for a Word document. This way you can fiddle with size in the .tex, but when you convert it to Word the figure looks good. If you produce directly the figure from R you can define a width and height in your code — e.g. pdf(..., width = X, height= Y). If you have a Mac you can open the image in Preview, go to Tools → Adjust size and modify the size directly. As a rule of thumb I find that using the width of the between-margins distance of a standard Word document works well — in my case that is 15.8cm, or 6.22 inches. If you get in the habit of producing images of an appropriate size as you go, this is not too much of a pain. And this way you can also set the DPI to have a figure resolution that makes you happy.

   - Error messages: Could not find image '...': go back to your .tex and make sure the extension is included in the image's file name, or if all your images have the same extension add the option `--default-image-extension=...`.

   - In-text references: Pandoc does not translate in-text references to Word (which is too bad as that is one of the coolest features of Latex)(that being said Word usually doesn't handle those well — but that's another story). What Pandoc does instead is copy the value of the label/ref literally in the Word document. For instance if you have in .tex "see Figure \ref{fig::dispersal}", which would render in the PDF as "see Figure 4" (and the 4 would be clickable), in the Pandoc translation .docx it will look like "see Figure [fig::dispersal]". Adding option `--parse-raw` removes the [...] labels as specified in the .tex's \label and \ref, which looks cleaner but gives no indication to your reader of the figure, table or section you are linking to. My suggestion is to leave the [...] labels in (i.e. not use the `--parse-raw` option) and warn the reader that they can search for the label using CTRL-F to find the figure, table or section that is being referred to.