Chapter 1
Testing HTML Export with make4ht

LianTze Lim

Abstract Y
our abstract. Ça va? (Yes accented characters work.)

1.1 Overview of Steps

$$MAE_t = \frac{1}{min(n', t)} \sum_{i=max(t-n'+1,1)}^{t} \|\hat{y}_i - y_i\|; \tag{1.1}$$

$$S(\omega) = 1.466\, H_s^2 \frac{\omega_0^5}{\omega^6} \exp\left[-3^{\frac{\omega}{\omega_0}}\right]^2 \tag{1.2}$$

$$S(\omega) = 1.466\, H_s^2 \frac{\omega_0^5}{\omega^6}\, e^{\left[-3^{\omega/(\omega_0)}\right]^2} \tag{1.3}$$

$$SA = Round\left(\left[1 - \frac{MAE(App)}{MAE(Median)}\right] * 100\right), \tag{1.4}$$

1. I tweaked the `latexmkrc` file so that `make4ht` is run along with `pdflatex`, so the HTML export is done only if you set your project to be compiled with `pdflatex`.
   (`make4ht` doesn't work well with `fontspec`; I haven't time to try this workflow with `XeLaTeX` nor `LuaLaTeX` yet. So let's just leave it at `pdflatex` first, yeah?)
2. `my.cfg` contains the settings I usually use in my workflow:
   - Output math as MathML, and render it in browsers using MathJax.
   - When `\includegraphics` uses `.jpg`, `.png`, `.jpg`, `.svg`, use the files directly for the HTML without conversion.

- When `\includegraphics` uses `.eps` and `.pdf`, convert them to `.png` using ImageMagick `convert`, and use those for the HTML.
- Tikz drawings are output as `.svg`.
- I've included some CSS styling. You can add a separate `.css` file for further styling.

3. When compilation is complete, use the steps at *View generated files* to download each generated file required. There's also a `allfiles.zip` that contains *all* generated files.

1.1.1 Caveats

- This is an experimental hackety hack – things may just not work! More a proof-of-concept rather than a stable solution on Overleaf at present.
- tex4ht doesn't work well with `fontspec` nor `authblk`.
- Avoid `\mathbf` – this broke MathML and MathJax for me.
- Avoid, or re-define the `multicol` environment to do nothing – tex4ht will really export text in two or three columns *by PDF page*, and it's not the most readable.

1.2 Introduction

Your introduction goes here!

1.3 Some LaTeX Examples

1.3.1 How to Include Figures

First you have to upload the image file (JPEG, PNG or PDF) from your computer to writeLaTeX using the upload link the project menu. Then use the includegraphics command to include it in your document. Use the figure environment and the caption command to add a number and a caption to your figure. See the code for Figure 1.1 in this section for an example.

Fig. 1.1 This frog was uploaded to Overleaf via the project menu. The `.jpg` file will be used as-is in the HTML export.

Fig. 1.2 PDF images will be converted to PNG when exported to HTML.

1.3.2 How to Make Tables

Use the table and tabular commands for basic tables — see Table 1.1, for example.

Item	Quantity
Widgets	42
Gadgets	13

Table 1.1 An example table.

1.3.3 How to Write Mathematics

LaTeX is great at typesetting mathematics. Let $X_1, X_2, \ldots, X_n$ be a sequence of independent and identically distributed random variables with $\mathrm{E}[X_i] = \mu$ and $\mathrm{Var}[X_i] = \sigma^2 < \infty$, and let

$$S_n = \frac{X_1 + X_2 + \cdots + X_n}{n} = \frac{1}{n}\sum_{i}^{n} X_i$$

denote their mean. Then as n approaches infinity, the random variables $\sqrt{n}(S_n - \mu)$ converge in distribution to a normal $\mathcal{N}(0, \sigma^2)$.

In this HTML export example, math is output as MathML, and will be rendered using MathJax.

1.3.4 How to Make Sections and Subsections

Use section and subsection commands to organize your document. LaTeX handles all the formatting and numbering automatically. Use ref and label commands for cross-references.

1.3.5 How to Make Lists

You can make lists with automatic numbering ...

1. Like this,
2. and like this.

. . . or bullet points . . .

- Like this,
- and like this.

. . . or with words and descriptions . . .

Word Definition
Concept Explanation
Idea Text

Testing some citations: [1, 2]

1.4 Tikz Drawings

TikZ drawings will be output as SVG, which should be rendered by most modern browsers.

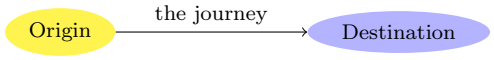

Fig. 1.3 TikZ drawings will be output as SVG, which should be rendered by most modern browsers.

References

[1] NTLK Project. Natural Language Toolkit (NLTK) 3.0 documentation, 2015.

[2] Francis Bond, Lian Tze Lim, Enya Kong Tang, and Hammam Riza. The Combined Wordnet Bahasa. *NUSA: Linguistic studies of languages in and around Indonesia*, 57:83–100, 2014.