[Pandoc][LaTeX][Python] LaTeX をはてなブログに変換する pandoc フィルタを作った

2020 年 8 月 24 日

目次

概要

最近, というか結構前からはてなブログの更新がめんどくさくなってきたので効率化をはかる. 一部の要素をはてな記法に変換し, tex ファイルからはてな記法対応 html を生成するフィルタを作ってみた.

R Markdown も内部で pandoc 使ってるので**そのうち R Studio から直接はてなブログに投稿できるようになる**……かもしれない

おおまかな機能は以下の通り.

- LaTeX コードを html + はてな記法に変換する
  - はてな記法は自分がよく使う機能のみ. html タグで簡単に代用できるものには対応していない
- `pandoc-crossref` との併用で図表や引用文献リストへのアンカーリンクによる相互参照も自動で生成
- 参照している画像ファイルをはてなフォトライフに自動アップロードする

大昔にtex ファイルをはてなブログ用に変換するスクリプトを書いた時は pandoc の仕組みを分かっておらず**構文解析器を無から作ろうとして挫折**したり**正規表現で誤魔化し**たりしていた. 実際には**フィルタという便利機能でだいたいなんとかなる**. ちゃんとマニュアルは読もう.

なお, このエントリも LyX で書いて変換したものをそのまま投稿している. 原稿はここ

1 使い方

1.1 インストール

インストール方法がやや込み入っている. まずは pandoc を用意する. Ubuntu のレポジトリだとバージョンが少し古いので github で .deb を落としてきてインストールする. 関連パッケージとの対応を考慮して最新版より少し古い 2.9.2.1 を使用した. このへんバージョン制約が多いみたいなので注意. Pandoc 2.9.2.1 に対応する pandoc-crossref の最新バージョンは 0.3.6.4 だが, これを少しだけいじったやつを使う (理由は後述). stack のほうは ver. 2.3.3

```
git clone git@github.com:Gedevan-Aleksizde/pandoc-crossref.git
cd pandoc-crossref
git checkout 4e02b07
stack clean
stack build
stack install
```

さらに pandoc-hateblo というのを作ってる人がいたのでフォークしてみたが「見出しを h3 タグに変える」という機能だけ作って数年前から止まっていいる. しかも私はまだ haskell がよく分かっていないので, フィルタに関する公式ガイドから辿って見つけた Python の panflute を使うことにした[*1]. その結果, オリジナルとは全く別物になった. なお pandoc を 2.9 にしたのは panflute が対応していない[*2]ためなので, この問題が解決されればもう少し簡単になる.

適当な場所にダウンロードしてパスを通しておく.

:embed:

```
git clone git@github.com:Gedevan-Aleksizde/pandoc-hateblo.git
git checkout hatena-filter
export PATH=<ここにpandoc-hateblo/binのパス>:${PATH} >> ~/.bashrc
```

フォトライフの API キーを取得して, 設定ファイル settings.json に書き込んでおく. blog_name は人によっては違う形式になるはず. FOTO_FOLDER は画像をアップロードしたいフォルダ. 未設定の場合は Hatena Blog になる

[*1] 内部的にはソースファイルを抽象構文木に変換, 抽象構文木にフィルタ適用, 抽象構文木から --to フォーマットへ変換, という順にパイプラインで実行されているのでフィルタはコマンドラインの標準入出力を操作できるスクリプトならなんでも良いらしい. そのうちでも Haskell か Lua で書いたほうが良いらしいが, 今回はどちらも構文を覚える時間が惜しかったので使わなかった. Haskell はそのうち覚えたいが……

[*2] https://github.com/sergiocorreia/panflute/issues/142

```
{
  "FOTO_API_KEY": "XXXXXX",
  "HATENA_USER": "YYYYY",
  "HATENA_BLOG": "YYYYY.hatenablog.com",
  "FOTO_FOLDER": "ZZZZ"
}
```

もしくは実行時に上記と同様の環境変数を設定する.

1.2 コマンド

変換したいファイルのディレクトリに移動し (フィルタの処理内でパスを参照するのがめんどくさくてやってないため) 以下のようなコマンドで変換する. `[...]` はオプション.

```
latex2hatena.sh [-o OUTPUT.html] [--bibliography=CITATIONS.bib] [--csl=CSL.csl] [-o OUT.html]
    INPUT.tex
```

複数ある場合 (1) カレントディレクトリの `settings.json`, (2) `hatena-filter/settings /` の `settings.json`, (3) 環境変数. で優先される. また, 画像ファイルのアップロードをせず動作確認だけしたい場合は `-M enable-upload=false` という引数を追加する. また, `-o ...` を省略すると標準出力に出る.

2 やりたいこと

以前と同様, LyX で書いた tex ファイルを**はてな記法** (はてな markdown ではない) に対応した html に変換したい. LyX は補完機能が充実しているので, 長いコマンドを短縮名のマクロで再定義するみたいなユーザー定義のマクロはあまり使わない, 使うとしてもコンテンツから分離したレイアウトの設定くらいにしか使わないという想定. 以下, 変換したいもの一覧 (参考: はてな記法一覧).

- `\hfref{}` は `[url:title=文字列]` の記法に変換.
  - ブログカード (? あれの呼び方がわからん) を埋め込みたい場合はとりあえずタイトルを ”`:embed:`“ にした場合のみ例外的に処理する.
  - タイトルを自動取得したい場合も同様に ”`:title:`“ にする.
  - ただしこの措置は同一ソースで複数種類の媒体へ変換するということができない (LaTeX 側が対応していない)
  - `\url{}` はそのまま URL を表示する. 例: https://ill-identified.hatenablog.com/
  - 平文で書いても自動でリンク
- 脚注もはてな記法に. `(())` で脚注にできるやつ
- 数式もはてな記法の `[tex: ...]` に置き換える.
  - qiita のようにいちいち変なエスケープをしなくて済むので簡単. 同様にエスケープが必要なはてな markdown ではなくはてな記法を使う理由の 1 つ.
  - いつからか別行立て数式が改行しなくなってしまったので別行立てで表示できるようにする.

  - タグを反映できるように `aligned` を全て `align` に置換
  - (pandoc 側でどっち使うのか指定できそうな気もするけどよくわからない)
- 引用ブロックは `>> ... <<` に変換した
  -「引用ブロック (字下げなし)」「詩句」(`quote` 環境) 「引用ブロック (字下げあり)」(`quotation` 環境), を変換
  - `\epligraph{}{}` も出典付き引用ブロックに変換
- コードブロックはスーパー pre 記法 (`>|| ... ||<`) に変換
  - `\begin{lstlisting}[language=]...\end{lstlisting}` で言語指定すれば反映
  - `minted` には未対応
- ただし出典付きの引用ブロックは未対応 (`>(出典)> (本文)<<` のやつ). pandoc が対応してない?
- 画像の挿入とキャプション周辺のレイアウト.
  - これははてなフォトライフとの連携も必要になるので厄介
- ちゃんと機能する図表・引用文献の相互参照
  - `pandoc-crossref` でアンカーリンク有効.
  - しかし `prettyref.sty` 前提で参照 ID は `fig:` や `tab:` の接頭辞が必要.
- 引用文献のフォーマット
  - `jecon.bst` は邦訳情報とかいろいろな拡張フィールドがあるが, CSL で作り直すのは大変
  - (up)BibTeX だと tex コードに変換してくれないので CSL で我慢する
  - 誰か BibLaTeX で書いて...
  - 引用文献にはアマゾンのリンクとかを自動で付けたい
- CSS やメタ情報は**未対応**
  - タグとかカスタム URL とかは手動で
  - CSS はグローバルな設定だけでなんとかする方針

3 解説

3.1 pandoc の内部処理の簡単な説明

このセクションの話はほぼ Pandoc の公式ドキュメント と Panflute の公式ガイドに書いてあることに基づく.

pandoc の内部での処理フローは, (1) ソースを解析する (2) 中間ファイルとして**抽象構文木** (ast) に変換する (3) 指定したフォーマットに再変換する, という処理をしており, フィルタは (2) の抽象構文木を修正する処理のことになる[*3]. なお中間ファイルは json 形式でも表現できるので, 多くの言語では json 前提で書いたほうが楽かもしれないが, Python には `pandocfilters` と `panflute` という 2 種類の pandoc フィルタ用モジュールがある. 今回は `panflute` を使ってみた.

Ast の詳細なドキュメントはまだない[*4]ため, 以下は私の解釈であり, 多少語弊があるかもしれない. しかし少なくともこの解釈を前提にこのフィルタを作っている.

[*3] https://pandoc.org/filters.html
[*4] https://github.com/jgm/pandoc/issues/3262

.ast ファイルに記述される抽象構文木の最小単位 (element) の書式は

```
<element_type> <attributes> <content>
```

となる. <element_type> が Str (文字列) とか Link (ハイパーリンク) とか, <attributes> は省略可能. 書式はパーレンで括ったリストになる.

```
("<id>", [<class>], [("<key>", "<value>"), (...)])
```

それぞれ HTML タグの ID, クラス, それ以外のスタイルや属性, になる. panflute では, <element>.identifier, <element>.classes, <element>.attributes でアクセスできる. 属性は collections.OrderedDict 型で, 平文 (Str) など属性を持たないタグもある. <content> には <element>.content でアクセスできる. この中には (1) ダブルクオーテーションで括った文字列, (2) 構文木の入れ子, (3) それらを [,] で括ったリスト, のどれかを与えられる. 構文木入れ子の最上層の型は Pandoc であり, <attributes> にメタ情報が入っている. ただし panflute ではこれは他のタグとは違う特別扱いで, doc.get_metadata() でアクセスする.

3.2 作った処理

まずは LaTeX で書いたこのブログの原稿 (実際には LyX で作成して tex ファイルとしてエクスポートしている) を pandoc で中間ファイルに変換する. まだ対応関係がよくわからんので以下を何度も実行しながら作った.

```
pandoc --mathjax -f latex -t native -o blog/intermediate.ast -F ./hateblo-filter.py  blog/test.
    tex
```

また, Python 内で panflute.convert_text() を使うと各フォーマットのテキストや panflute のオブジェクトを相互に変換できるのでいろいろ確認できる (たまにうまくいかない).

要素単位の変換処理は panflute でだいたいできるが, 面倒なのは相互参照である. 例えば本文中の「図 1」「表 2」とか「式 (3)」みたいな通し番号を自動で割り当てた上で, アンカーリンクを付けたいというのが要件である. 今回使用している pandoc は図表の相互参照に対して自動で番号を割り当ててくれる. しかし図や表側のキャプションに「図 XX: ...」のような番号を振ることもない[*5]. また, 文献引用も --bibliography= にファイルを指定すれば自動で pandoc-citeproc が呼び出されるが, これも相互参照にアンカーリンクを貼ってくれない.

これに対して pandoc-corssref という相互参照用のフィルタがある. それぞれ末尾参考文献リストと図表の相互参照にアンカーを張るのことができる[*6]. このフィルタは基本的に Markdown から他の媒体への変換を想定しており, LaTeX からの変換は想定していない (issue #250) ことになっているが, 今回インストールしたバージョンは LaTeX からでも一応動作するようだ.

[*5] 加えて, 相互参照として認識されるのは \ref{...} コマンドのみで prettyref.sty や refstyle.sty で提供されるコマンドには対応していない.

[*6] この機能は link-citations=true と linkReferences=true をメタデータに追加することで切り替えられる. しかしなんでハイフンだったりキャメルだったりするのか? また, デフォルトでは false のとのことだが, 想定してない LaTeX からだからなのか指定しなくてもリンクされる.

ただしいくつか問題がある．まず，pandoc-crossref の相互参照 ID の命名にはルールがある．図なら fig:，表なら tbl:，セクションタイトルなら sec: と言うふうに接頭辞が必要だ．これは私が LyX で使っている prettyref.sty でも似たルールで採用されている．しかし，こちらでは表は tab: であり，pandoc-crossref 側は接頭辞がハードコードされている．ver. 0.4 以降では自由にカスタマイズできるようにするらしい[*7]が，今は pandoc, panflute との互換性で古いバージョンを使わざるをえない．そこで**ハードコードにはハードコードの上書きで対処**した．全くスマートではないがまだ haskell がわからないので一旦これでやっていく．これがオリジナルではなくフォークリポジトリを用意した理由である．もう 1 つは LaTeX からの変換の場合，参照 ID が表示されてしまう点．これは panflute でその要素を除去するように修正する．

それ以外の置き換えはさほど難しい話ではなく，panflute で簡単に対処できる．

最後に，はてなブログに画像を貼り付ける場合，よそのサイトからの直接リンク以外は「はてなフォトライフ」というサービスに投稿した画像しか使えない．このサービスには Web API が用意されている[*8]ので，これを利用して自動でアップロードする機能も付けている．

あとは pandoc のオプションでだいたいなんとかなる．以下のようなオプションの想定で実行している．

```
pandoc --mathjax --wrap=auto -f latex -t html  -F pandoc-crossref -F pandoc-citeproc -F hateblo-
    filter.py -MfigureTitle='図' -MfigPrefix='図' -MtableTitle='表' -MtblPrefix='表' --
    bibliography=CITATIONS.bib --csl=CSL.csl -o blog/OUTPUT.html INPUT.tex
```

4　用例

このセクションでは上記で挙げた機能を確認する．

はてな記法のリンクはリンク先のタイトルを取得できたり埋め込みできたりと便利である．たとえば [https://hogehoge:title] で『:title:』のようにタイトルを自動取得できる．:embed: を使えば「ブログカード」を生成する．

:embed:

pandoc はデフォルトだと以下のような文中の別行立て数式を，

$$\sin^2 x + \cos^2 x = 1, \tag{A}$$

$$\int_{\mathbb{R}} x dF(x;\theta) = \mu \tag{B}$$

のように独立行で挿入すると改行されずインラインになってしまう．一方で明示的に改行すると段落まで改められてしまう．そこで改行コードを直接挿入することにした[*9]．また，mathjax を指定すると数式を \[... \] で囲むが，はてな記法 [tex: ...] の方が便利なので置換している．

引用ブロックもはてな記法が便利なので，quote, quotation, epigraph 環境を変換対象とする[*10]．

> かたつぶりそろそろ登れ富士の山

[*7] https://github.com/lierdakil/pandoc-crossref/issues/200

[*8] http://developer.hatena.ne.jp/ja/documents/fotolife/apis/atom

[*9] 最近のバージョンはどういうルールや意図で改行の有無を決めているのかよくわからない．–wrap=preserve なら SoftBreak で改行してくれるが，一方でソースファイルで改行も何も無いところに唐突に SoftBreak を挿入してしまうのでレイアウトが崩れるし，かといって none や auto にすると全然改行してくれない．

[*10] ただし，はてな記法の引用ブロックの出典には URL を指定することしかできないため，\LaTeX との互換性は完全ではない．

	1	2
a	x	o
b	o	x

表1　ここに表のキャプション

小林一茶

`\texttt{epigraph}` で出典付き引用

> 皆さんは最近、文中に「メキシコ」「真の男」「バンデラス」「腰抜け」などのワードが頻出する怪文書やネットミームをご覧になった事があるでしょうか？　それに伴い「逆噴射文体」「逆噴射先生」等の呼称を見かけて、ますます混乱した事は？　これは、逆噴射聡一郎先生のシグネチャー文章です。（あるいは、それを模倣したファンによる文体模写です）
>
> ---
>
> https://diehardtales.com/n/n73ec21c8457b:title= ダイハードテイルズ所属作家紹介：逆噴射聡一郎とは？

4.1　相互参照

図表のキャプションと相互参照. 図 1, 表 1 のように付番とアンカーリンクを自動で行う.

数式参照も可能. A, B

文献引用にも対応. これらの本を表示例に選んだ理由は特にない: 星野 (2009), 山本 (2019), Hastie et al. (2009), Igami (2017)

5　未解決事項

- 目次の配置は自動化していない. pandoc の `--toc` オプションは `-s` と併用しない限り意味がない. しかし `-s` では不要なヘッダが大量に作られてしまい邪魔である.
- 数式参照には数式側に `\tag{}` が必須
- `prettyref/refstyle` への対応. 相互参照のときに参照 ID だけ書けば十分なはずだが, `\ref{}` では「図」とか「表」とか接頭辞も毎回いちいち書く必要がある. これはナンセンスなので私は LyX 上では `prettyref.sty` の書式設定で対処している (LyX 開発者側は類似機能を持つ `refstyle.sty` に移行したがっているが私は未対応). しかし今回はまだ pandoc は `prettyref.sty` の `\prettyref{}` に対応していない.
- バックスラッシュで始まる TeX コマンドを平文で書くと MathJaX が暴発することがある. しかしこれははてなブログが悪いのでこちらでは解決できない.
- 参考文献リストのフォーマット. CSL ファイルか BibLaTeX のフォーマットを書く必要がある. CSL は XML で書くので比較的簡単だが, 仕様上, 私が LaTeX でいつも使っている `jecon.bst` を再現するのは難しい. おそらくは BibLaTeX で書いたほうがよそにも流用できて良さそうで (Word を使う予定がないので CSL は全く出番がない), あるいはフィルタとして処理を書くという方針でもできるだろうが, どちらも大変なので一旦保留.
- せっかく API 使うのなら投稿まで自動化してもよかったけどめんどくさくなった

図1　ここにキャプション

- 参考文献が Amazon にあるなら, リンクを自動で追加したい

参考文献

Hastie, Trevor, Robert Tibshriani, and Jerome Friedman (2009) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*: Springer, 2nd edition, retrieved from *here*, （井尻善久・井出剛・岩田具治他訳，杉山将・井手剛・神嶌敏弘・栗田多喜夫・前田英作監訳，『統計的学習の基礎 ―データマイニング・推論・予測―』，共立出版，2014 年,）.

Igami, Mitsuru (2017) "Estimating the Innovator's Dilemma: Structural Analysis of Creative Destruction in the Hard Disk Drive Industry, 1981–1998," *Journal of Political Economy*, Vol. 125, No. 3, pp. 798–847, June, DOI: 10.1086/691524.

星野崇宏 (2009) 『調査観察データの統計科学 –因果推論・選択バイアス・データ融合』，岩波書店.

山本勲 (2019) 『人工知能と経済』，勁草書房.