Getting Started with ContentMine table processing software

The ContentMine document processing system is known as norma. This document introduces the commands used to transform tables to semantic format.

System requirements

- Java 1.8 runtime (Oracle or OpenJDK)
- OS: Linux, MacOS X

The following instructions assume a command-line shell such as Linux or MacOS X.

Windows is not currently supported for this version.

Version

This document is intended for use with the current snapshot release `norma-0.4.1-SNAPSHOT`

Running norma on an existing corpus

To get the binary and the current corpus (the Open Access subset of the CM-UCL-II corpus), either:

If you have git installed, do:

```
git clone https://github.com/ContentMine/cm-uclii/
```

or download the following and and unzip it:

https://github.com/ContentMine/cm-uclii/archive/master.zip

This repo has sub-directories:

- **`corpus-oa-uclii-01`** -- a directory in CProject format, containing the tables manually extracted from the OA papers and converted into ContentMine SVG
- **`norma-20171212`**

and also contains shell scripts:

- **`transform.sh`**
- **`cleancproject.sh`**

`norma-20171212/bin/norma` is the latest norma binary (version `norma-0.4.1-SNAPSHOT`)

Usage

Convert tables from SVG to HTML

The core functionality for the CM-UCL project is to convert the tables which have been converted to SVG into HTML.

To run this conversion process on the Open Access papers from the CM-UCLII corpus, open a terminal. Go to the the directory `cm-uclii` and use the following command line:

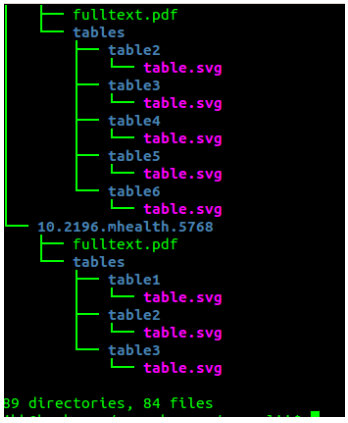
```
./bin/norma --project corpus-oa-uclii-01 --fileFilter
"^.*tables/table(\\d+)/table( \\d+)?\\.svg"  --outputDir corpus-oa-
uclii-01 --transform svgtable2html
```

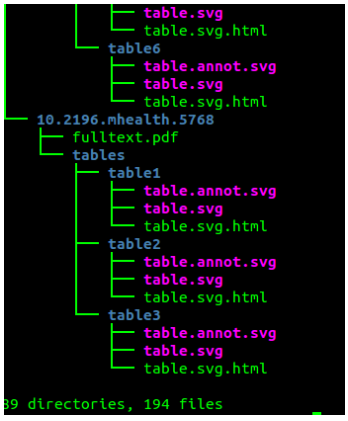
This will generate HTML5 files stored under the `corpus-oa-uclii-01` directory tree.

The name of each document is output to the command line as norma runs. More detailed output from the run including any errors are logged to `norma.log` in the current directory. Each day's log is rolled over into a dated log file so that the current day's log does not grow too large.

Expected behaviour – details

A successful run will output resulting HTML files (table.svg.html) and some intermediate files capturing the system's analysis of the table (e.g., table.annot.svg, a version of the original table.svg with annotations indicating its structure and other results of layout analysis).





Creating an HTML demo page of the whole corpus

Additional bash scripts are provided for convenience to create a browsable HTML page to view the results across the whole corpus.

Before using the scripts, set the `NORMA` environment variable to point to the full absolute path to the location of the main norma executable, e.g., `/home/abc/work/cm-uclii/norma-20171212/bin/norma`

To create a browsable HTML page showing styled tables, run:

```
transform.sh corpus-oa-uclii-01
```

To view the browsable page open a browser (e.g., Firefox or Chrome and enter the following file URL, replacing `/home/abc/work` with the path to the directory containing of the `cm-uclii` directory):

`file:///home/abc/work/cm-uclii/corpus-oa-uclii-01/tableViewList.html`

Example outputs

A example of a table whose structure has been resolved to identify subtables, super-column headers and columns with compound contents split into individual numerical values. Note that the colours and styling for this demo HTML is provided by using an external CSS3 stylesheet which references the semantic markup in the HTML5 file.

For DOI: 10.1093.alcalc.ags133 Table 4:

10.1093.alcalc.ags133

table1 table2 table3 table4 table5

Table 4. Moderating effects on heavy drinking and frequency of binge drinking using logistic regression (intention-to-treat analysis) at the 1- and 6-month follow-up

	1-month follow-up			6-month follow-up			1-month follow-up		6-month follow-up	
	OR	95% CI	P	OR	95% CI	P	95% CI :0	95% CI :1	95% CI :0	95% CI :1
Heavy drinking by										
Gender	0.72	[0.35–1.44]	0.35	1.39	[0.78–2.49]	0.27	0.35	1.44	0.78	2.49
Readiness to change T0	0.61	[0.26–1.48]	0.28	1.28	[0.64–2.55]	0.49	0.26	1.48	0.64	2.55
Problem drinking T0	0.75	[0.35–1.65]	0.48	0.74	[0.41–1.33]	0.31	0.35	1.65	0.41	1.33
Freshmen T0	1.04	[0.45–2.38]	0.93	1.15	[0.57–2.33]	0.69	0.45	2.38	0.57	2.33
Fraternity or sorority membership T0	1.07	[0.53–2.18]	0.85	0.87	[0.48–1.56]	0.63	0.53	2.18	0.48	1.56
Carnival participation T1	0.78	[0.31–1.94]	0.59	0.99	[0.55–1.80]	0.99	0.31	1.94	0.55	1.80
Frequency of binge drinking by										
Gender	0.75	[0.37–1.52]	0.43	1.38	[0.78–2.44]	0.27	0.37	1.52	0.78	2.44
Readiness to change T0	0.67	[0.28–1.61]	0.37	1.30	[0.65–2.59]	0.45	0.28	1.61	0.65	2.59
Problem drinking T0	0.70	[0.32–1.49]	0.35	0.75	[0.41–1.36]	0.33	0.32	1.49	0.41	1.36
Freshmen T0	1.00	[0.45–2.24]	0.99	1.14	[0.57–2.32]	0.71	0.45	2.24	0.57	2.32
Fraternity or sorority membership T0	1.08	[0.54–2.18]	0.83	0.90	[0.50–1.61]	0.72	0.54	2.18	0.50	1.61
Carnival participation T1	0.83	[0.33–2.09]	0.69	1.06	[0.59–1.91]	0.84	0.33	2.09	0.59	1.91

Note. T0, baseline assessment; T1, 1-month follow-up.

corpus-oa-uclii-01/10.1093.alcalc.ags133/tables/table4/table.svg.html

Subtables are indicated by dark green headings and pale green background. Additional columns generated by splitting compound column contents are coloured yellow, or yellow-green for values within subtables. Row headings (observation labels) are orange within subtables, red for top-level rows.

Generating new results for the current corpus

To regenerate the results and browsable demo HTML, first clean the Cproject directory tree.

To remove all files generated by norma from the corpus's Cproject, run the following :

```
cleancproject.sh corpus-oa-uclii-01
```

Then run the `transform.sh` script which will generate the main HTML and CSV outputs and also the browsable HTML demo pages:

```
transform.sh corpus-oa-uclii-01
```

Adding new tables from PDF documents

This norma binary can also be used to convert tables in new papers from PDF to HTML/CSV tables, in conjunction with the use of some manual steps in an SVG editor, such as Inkscape.

Extracting new tables for use by norma involves the following steps:

1. Create the directory structure for extracted and converted files

First, the user needs to organize all original PDFs into one folder. Second, this folder needs to be converted to a cproject structure. The cproject structure normalizes the contents for each paper into a ctree, such that subsequent operations are trivial to standardize (and extensions can be applied relatively easily). For example, the root folder might contain ctree1.pdf, but after transforming the root folder into a cproject it contains a folder ctree1/ with fulltext.pdf. By running the command

```
./bin/norma --project corpus-oa-uclii-01  --fileFilter '.*/(.*).pdf' --makeProject '(\1)/fulltext.pdf'
```

the folder corpus-test2 (--project corpus-test2) is restructured into a cproject structure, containing a folder for each PDF file (--fileFilter '.*/(.*).pdf' --makeProject '(\1)/fulltext.pdf'). This results in the following folder structure:

```
cproject/
├── ctree1
│   └── fulltext.pdf
├── ctree2
│   └── fulltext.pdf
...
└── ctreeN
    └── fulltext.pdf
```

After converting the folder into a cproject, the norma software is applied to convert the PDF files into separate SVGs per page. In order to convert each page of the PDF into a separate SVG file, we used the following command

```
 ./bin/norma --project corpus-test2 -i fulltext.pdf --outputDir corpus-test2 --transform pdf2svg
```

resulting in a `svg/` folder for each ctree in the structure presented above. That is, each ctree now contains a folder with one vector file for each page in the fulltext PDF.

The following step, extracting the tables from the page and saving these, currently needs to be done manually. We recommend using the FOSS software Inkscape to do this. For each article, open the pages containing tables, select the area of the table, and press the keyboard shortcut SHIFT+1 (i.e., !) to invert the selection; then press the delete key to delete everything except the table. Then save the whole page with the table in its original position into a file as Plain SVG (not Inkscape SVG) and structure the folders as follows:

```
cproject/
└── ctree1
    ├── fulltext.pdf
    └── tables/
        └── table1/
            └── table.svg
        └── table2/
            └── table.svg
```

where table1 contains the first table, table2/ contains the second table, etc. Note that the number suffixes need to be unique but do not need to correspond to the table numbering used in the document, which may include examples like 'Table IV', 'Table 2 (cont.)' etc.

Finally, each table is converted to a data file with norma. The following command

produces an structured HTML5 file

```
./bin/norma --project corpus-oa-uclii-01 --fileFilter
"^.*tables/table(\\d+)/table(_\\d+)?\\.svg" --outputDir corpus-oa-uclii-
01 --transform svgtable2html
```

Adding new tables

To run the conversion process on tables in new documents, the following 5 steps are needed:

1. Create directory structure to hold results of conversion and extraction for each document
2. Copy the original PDF into the directory structure
3. Run ContentMine pdf2svg transform using norma binary
4. Manually remove all irrelevant content from each SVG page to leave only the table, in its original position in the page.
5. Run the table-conversion process svgtable2html using ContentMine norma as for the UCL-II corpus in GitHub.

Details

1. Create the directory structure to hold document and files which will be generated as part of the process. This will hold all files including pre-processed table files and the results of automatic conversion in HTML and CSV.

```
cproject/
└── ctree1
    └── fulltext.pdf
    └── tables/
        └── table1/
            └── table.svg
        └── table2/
            └── table.svg
```

2. Copy the PDF document into the directory structure. We use the convention of naming this `fulltext.pdf`

3. Run the following command on the PDFs in the CTree:

```
java -jar bin/norma-0.5.0-SNAPSHOT-jar-with-dependencies.jar --project
corpus-test2 -i fulltext.pdf --outputDir corpus-test2 --transform
pdf2svg
```

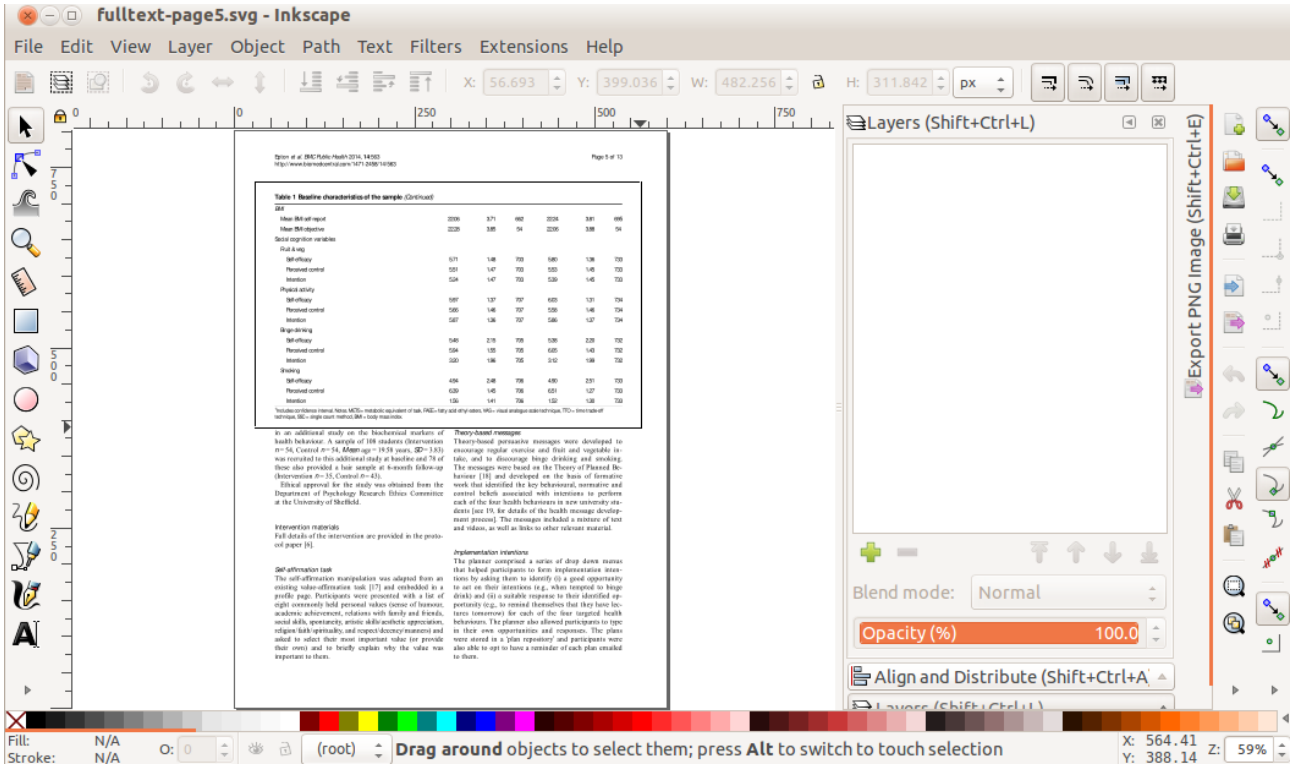
This will result in an SVG file for each page in each PDF. These SVG files will be generated under the svg/ subdirectory of the ctree structure:

```
cproject/
└── ctree1
    └── fulltext.pdf
    └── tables/
```

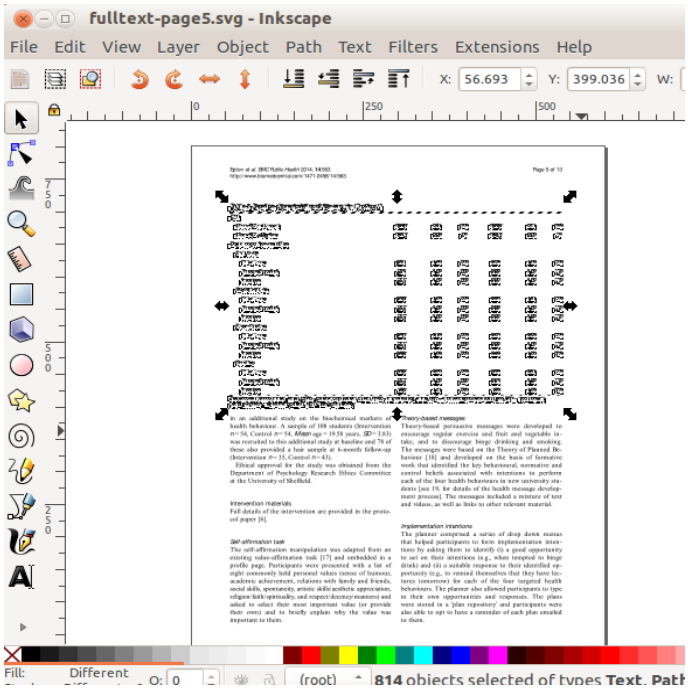
```
        └── table1/
            └── table.svg
        └── table2/
            └── table.svg
```

4. Manual step. Using an SVG editor, such as Inkscape (https://inkscape.org/), edit each relevant page as follows.

Select the table within the page in the usual way by dragging the selection cursor:

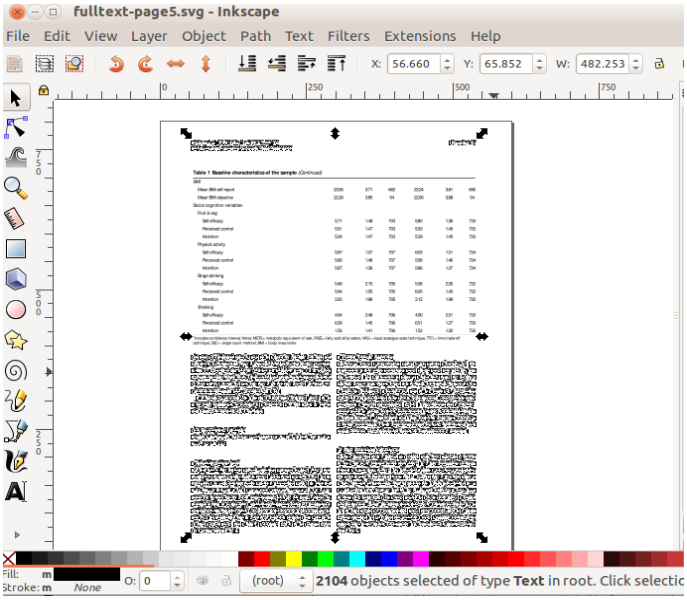


Wait until all objects such as characters and lines in the deselected area are selected. In a complex page this may take a few seconds. When this process is complete the selected table should look as follows:



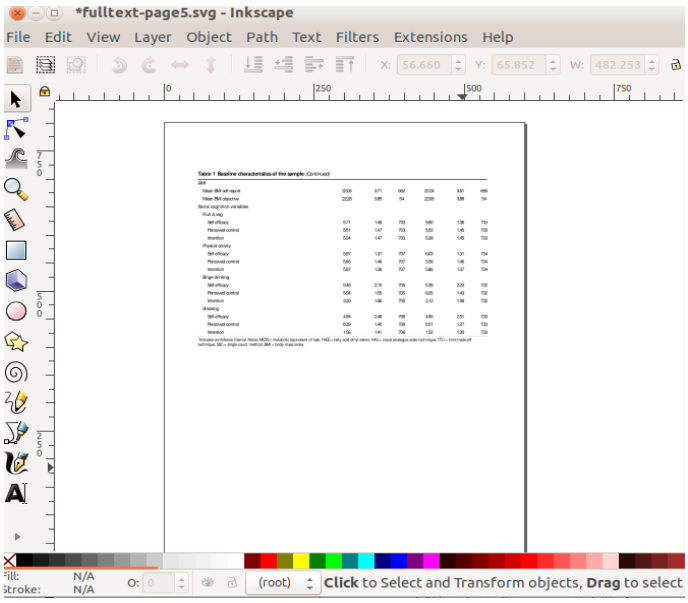
We now need the table to be the only object on the page. We do this by delete everything apart from the selected table.

To delete the non-table content, with the table selected as above, invert the selection by typing `!` (or `SHIFT+1`) to select the non-table content instead. Wait until all objects such as characters and lines in the deselected area are selected. In a complex page this may take up to 10 seconds.



When the selection inversion process is complete then type `CTRL-X` or `DELETE` to remove the non-table content.

The finished page should contain only the table:



Finally save the page as SVG

Choose output format `Plain .SVG`. Note: do not use the default output format `Inkscape .SVG`

Save the file into the CTree as `tableN/table.svg` where N is unique within the Ctree for this PDF document.

Note that there should only be one table per SVG file. Where multiple tables appear on the same page, repeat the procedure once for each to produce separate `tableN/table.svg` files.

The resulting ctree should have form

```
cproject1/
  ctree1/
    fulltext.pdf
    svg/
    tables/
      table1/
        table.svg
      table2/
        table.svg
  ctree2/
    fulltext.pdf
    svg/
    tables/
      table1/
        table.svg
...
```

10.

5. Run ContentMine norma over the cproject as before using `transform.sh` or a the full command line.