

Tool to Extract Titles and Footnotes from TFL Shell

Wenrui Qi, Sanofi (Shanghai) Co., Ltd.

ABSTRACT

In the pharmaceutical industry, generating accurate and well-formatted statistical tables, figures and listings (TFLs) is a critical task for SAS programmers. One of the most time-consuming works is extracting table titles and footnotes from statistical shells, which often requires manual intervention and increases the risk of errors. To address this issue, we have developed an automated tool using Python that efficiently extracts table titles and footnotes from statistical shells and integrates them into an excel file to support the generation and packaging of TFLs.

This paper will introduce the application scenarios and development process of the tool, demonstrate its functionality with real-world examples, and discuss its potential impact on improving efficiency and consistency in statistical reporting. Specific examples will be shared.

INTRODUCTION

In the pharmaceutical and clinical research industries, the generation of statistical TFLs is a fundamental component of data analysis and regulatory submissions. TFLs must adhere to strict formatting guidelines and include essential contextual information, such as title, program name, filename, footnote and so on. Traditionally, Statistical programmers extract metadata from TFL shells manually. However, these methods are often labor-intensive, prone to human error, and can lead to inconsistencies in outputs, especially when working with large datasets or tight deadlines. It is imperative to automate this part of workflow.

This tool leverages Python's powerful text processing libraries, such as regular expressions and string manipulation, to parse and extract relevant metadata with high precision. All titles and footnotes of each TFL outputs can be extracted automatically from TFL shell and fill them into an Excel file. By automating this step, the tool significantly reduces manual effort, minimizes human error, and enhances overall productivity in report generation. Powerful libraries Python-docx, Pandas are used in this tool to extract essential information in TFL shells.

PROCESS OF GENERATING TFLS

Different corporation has different process to generate TFLs. In SANOFI, the process basically follows 4 steps (**Figure 1**):

1. Statisticians provide TFL shell for Statistical programmers.
2. Statistical programmers extract essential contextual information from TFL shell and summarize them in refflist.xlsx and refflist.csv.
3. Information in refflist.csv is transformed into SAS dataset (gs_refflist.sas7bdat).
4. Read gs_refflist.sas7bdat in SAS and generate TFLs (Reporting.rtf).

This tool aims to enhance the efficiency in step 2: extract table titles and footnotes automatically from TFL shells and integrate them into refflist.xlsx.

1. Titles of each TFL and headings of each level of section are recorded in the table of contents (**Figure 3**).
2. For each level a heading or title is demoted, the corresponding indentation in the table of contents is increased by one level (**Figure 3**).
3. Unify the font and format of footnotes and incorporate them into the table along with the main content of the TFL. The footnote section should be placed in the last cell of the table. Table borders are not displayed (**Figure 4**).
4. Each footnote should be on its own line (**Figure 4**).

16.2.4 Demographic data, data at baseline and medication details	32
Demographics and baseline characteristics	32
Demographics 1	32
Medical/Surgical history and medical findings	35
Medical 1	35

Figure 3. Standard table of content in TFL shell

Pre-specified medical or Surgical history - Randomized population			
n (%)	Trt A (N=##)	Trt B (N=##)	All (N=##)
Any medical or surgical history	## (##.%)	## (##.%)	## (##.%)
XXX	## (##.%)	## (##.%)	## (##.%)
XXX	## (##.%)	## (##.%)	## (##.%)
XXX	## (##.%)	## (##.%)	## (##.%)
...	## (##.%)	## (##.%)	## (##.%)
Footnote 1 Footnote 2 Footnote 3			

Footnotes are placed in the last cell of the table



Borders are not displayed

Pre-specified medical or Surgical history - Randomized population			
n (%)	Trt A (N=##)	Trt B (N=##)	All (N=##)
Any medical or surgical history	## (##.%)	## (##.%)	## (##.%)
XXX	## (##.%)	## (##.%)	## (##.%)
XXX	## (##.%)	## (##.%)	## (##.%)
XXX	## (##.%)	## (##.%)	## (##.%)
...	## (##.%)	## (##.%)	## (##.%)
Footnote 1 Footnote 2 Footnote 3			

Figure 4. Standard layout of footnote in TFL shell

GENERAL WORKFLOW

This tool is generated by Python. Only TFL shell need to be prepared for using this tool. There are 5 steps to extracts table titles and footnotes from TFL shells and integrates them into reflat.xlsx:

1. Iterate table of content in TFL shell.
2. According to the indentation, place headings/titles of different levels into different columns in title.xlsx.
3. Read column tit_CSR from title.xlsx. Based on the TFL names in this column, fetch the corresponding footnote for each TFL from the main content of the TFL shell.
4. Place each footnote sequentially from top to bottom into different columns of footnote.xlsx.
5. Combine title.xlsx and footnote.xlsx together into reflat.xlsx.

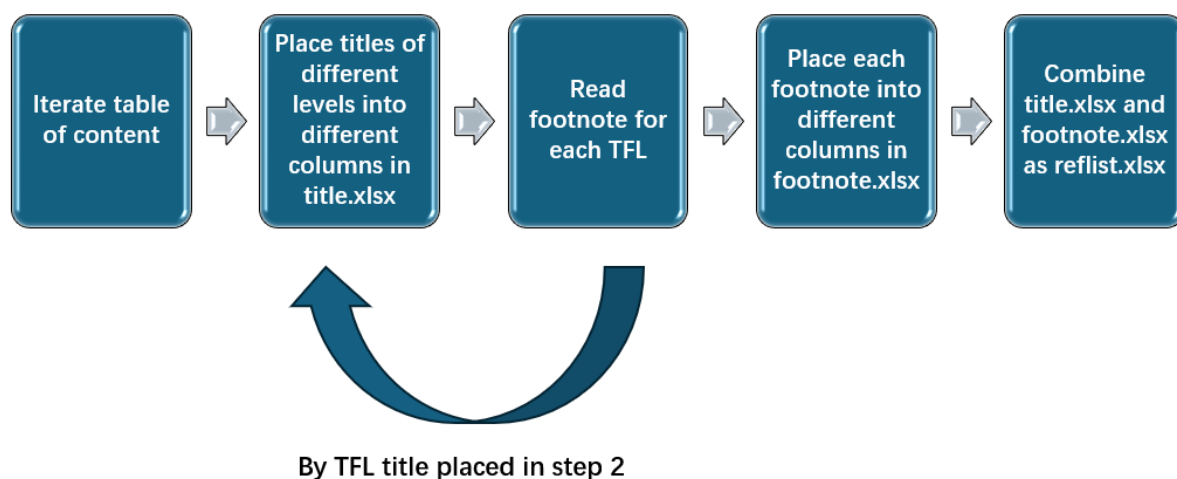


Figure 5. General workflow of the tool

SPECIFIC PART OF THE TOOL

TITLE PART

Titles are extracted by iterating table of content in TFL shell. Two parts are generated in reflist.xlsx: TFL titles and headings of different level. TFL titles include titles of specific tables, figures and listings. They are generated under column tit_CSR (**Figure 7**). Headings of different level are placed sequentially from high level to low level under column tit1_APP to titX_APP (value of X based on how many levels of heading and title does TFL shell have). For example, if there are only one section 16.2.4 in a shell (**Figure 6**). It has two level of headings and their corresponding titles. In this case, X of titX_APP is equal to 3.

Furthermore, for low-level headings or specific TFLs that correspond to the same high-level heading, the high-level heading remains consistent. For example, there are three tables under high-level heading “Prior medications” in TFL shell of **Figure 6**. For these tables, they share the common tit2_APP “Prior medication” (**Figure 7**).

Finally, both titles and headings are extracted into excel file title.xlsx temporarily.

16.2.4 Demographic data, data at baseline and medication details	32
Demographics and baseline characteristics	32
Demographics 1	32
Medical/Surgical history and medical findings	35
Medical 1	35
Prior medications	37
Prior medications 1	37
Prior medication 2	38
Prior medications 3	39
Concomitant medications	40
Concomitant medications 1	40
Concomitant medications 2	41
Concomitant medications 3	42

The diagram illustrates the relationship between TFL Titles and Headings of different level. A red arrow points from 'TFL Titles' to a table with columns tit_CSR, tit1_APP, tit2_APP, and tit3_APP. A blue arrow points from 'Headings of different level' to the same table. The table contains 10 rows of data, with the first row having a rowspan of 10 for tit1_APP.

tit_CSR	tit1_APP	tit2_APP	tit3_APP
Demographics 1	16.2.4 Demographic data, data at baseline and medication details	Demographics and baseline characteristics	Demographics 1
Medical 1	16.2.4 Demographic data, data at baseline and medication details	Medical/Surgical history and medical findings	Medical 1
Prior medications 1	16.2.4 Demographic data, data at baseline and medication details	Prior medications	Prior medications 1
Prior medication 2	16.2.4 Demographic data, data at baseline and medication details	Prior medications	Prior medication 2
Prior medications 3	16.2.4 Demographic data, data at baseline and medication details	Prior medications	Prior medications 3
Concomitant medications 1	16.2.4 Demographic data, data at baseline and medication details	Concomitant medications	Concomitant medications 1
Concomitant medications 2	16.2.4 Demographic data, data at baseline and medication details	Concomitant medications	Concomitant medications 2
Concomitant medications 3	16.2.4 Demographic data, data at baseline and medication details	Concomitant medications	Concomitant medications 3

FOOTNOTE PART

Footnote1: BMI: Body mass index↵
Footnote2: height↵
Footnote3: weight↵

A	B	C	D
tit_CSR	foot1	foot2	foot3
Demographics 1	Footnote1: BMI: Body mass ind	Footnote2: height	Footnote3: weight

FINAL CONBINATION OF REFLIST.XLSX

A	B	C	D	E	F	G
tit_CSR	tit1_APP	tit2_APP	tit3_APP	foot1	foot2	foot3
Demographics 1	16.2.4 Demographic data, data at baseline and medication details	Demographics and baseline characteristics	Demographics 1	Footnote1: BMI: Bod	Footnote2: height	Footnote3: weight
Medical 1	16.2.4 Demographic data, data at baseline and medication details	Medical/Surgical history and medical findings	Medical 1	a Medical or surgicn (%) = number and	Note: A <participant	
Prior medications 1	16.2.4 Demographic data, data at baseline and medication details	Prior medications	Prior medications 1	WHO-DD [dictionary n (%) = number and	Note: As per proto	
Prior medication 2	16.2.4 Demographic data, data at baseline and medication details	Prior medications	Prior medication 2	WHO-DD [dictionary n (%) = number and	Note: As per proto	
Prior medications 3	16.2.4 Demographic data, data at baseline and medication details	Prior medications	Prior medications 3	WHO-DD [dictionary n (%) = number and	Note: As per proto	

5

UI OF THIS TOOL

After saving the Python script of this tool, local TFL shell file can be uploaded to an interactive UI web page in your local environment. The web page will automatically generate three Excel files: title.xlsx, footnote.xlsx, and refile.xlsx. The steps are as follows:

1. Save the tool's Python script to desired local folder. Open the command prompt (cmd), navigate to that directory, and then run the following command to launch the web application:

```
Bash

python -m streamlit run name_of_tool_script.py
```

Figure 11. Standard command in command prompt (cmd)

Once you run the command, the tool will start and open the web interface in your default browser. For example, if tool's Python script "pharmasug_tool" is saved under folder "C:\pharmsug\tool", then command in command prompt (cmd) is:

```
C:\pharmsug\tool>python -m streamlit run pharmasug_tool.py
```

Figure 12. Example command in command prompt (cmd)

2. Upload local TFL shell file to the UI web. Then click "start".

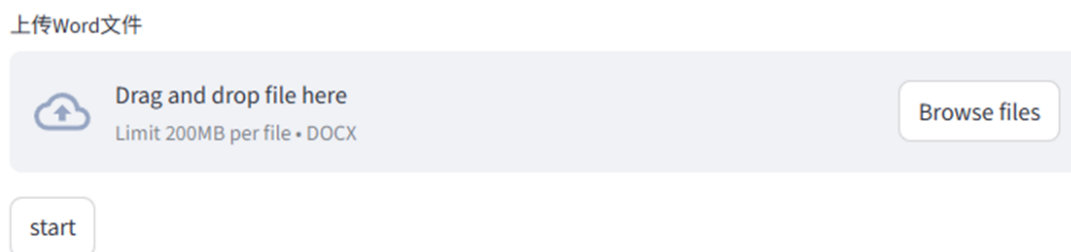


Figure 13. UI web to upload local TFL shell

3. Title.xlsx, footnote.xlsx, and refile.xlsx are generated. Excel files can be downloaded as needed.

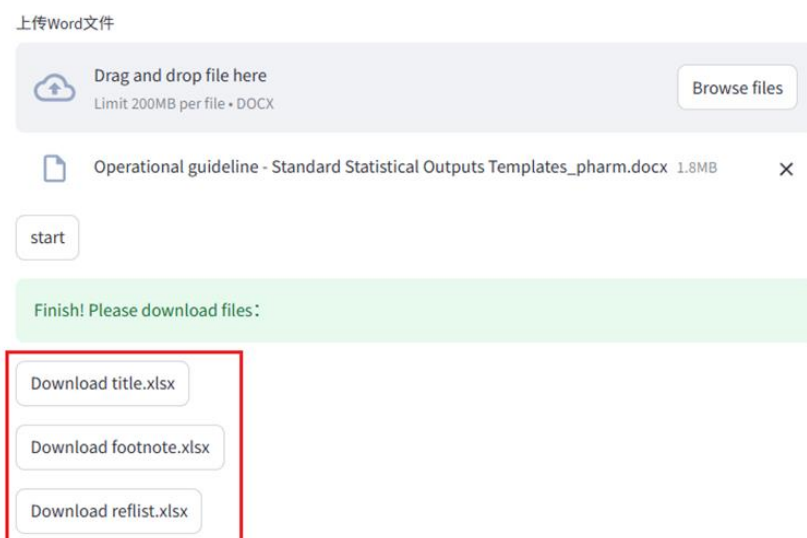


Figure 14. UI web to download excel files

LIMITATIONS

1. This tool has very strict format requirements for the TFL shell. If the format of table of contents or footnotes does not match the description in the **PREPARATION BEFORE USING THIS TOOL** section, the tool will not be able to extract titles or footnotes accurately.
2. Each heading and TFL title must appear in the table of contents of the TFL shell. If multiple similar TFLs correspond to the same table template in the shell, manual copying and pasting can be performed in the Excel file (reflist.xlsx).
3. This tool is only applicable to the TFL generation process following the **Process of Generating TFLs** section. When generating TFL outputs, titles and footnotes must be read from an Excel file (such as reflist.xlsx).

CONCLUSION

In this paper, we have introduced an innovative tool developed using Python that automates the extraction of table titles and footnotes from statistical shells. Although this tool has several limitations, it can help to enhance efficacy and accuracy. By ensuring strict adherence to format requirements, our tool significantly reduces manual effort and minimizes human error.

Furthermore, automation is a trend in statistical analysis of clinical trial. There is a tremendous potential for automation in many aspects in our daily work. By leveraging tools like the one described in this paper, we can significantly enhance efficiency, reduce human error, and free up valuable time for more complex analytical tasks. Looking forward to the emergence of more automated tools in the future.

REFERENCES

Wes Mckinney, 2018.6, Python for Data Analysis: Data Wrangling with Pandas, Numpy, and IPython, 2nd Edition Chinese version, Beijing, Machinery Industry Press.

Weiwei Zhang, "Extracting Titles and Footnotes from TLF SHELL with PYTHON", PharmaSUG China 2022, Paper 115 – AD

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Ray Qi
Sanofi
Ray.qi@sanofi.com