PharmaSUG 2021 - Paper AD-054

How to Translate RTF Documents

Jundong Ma, Dizal Pharma, Shanghai, China
Zhiping Yan, Dizal Pharma, Beijing, China

ABSTRACT

Conducting clinical trials across multiple countries is becoming increasingly common for clinical research. SPs (Statistical Programmers) may need to produce multiple packages of TLFs (*tables*, *listings*, and *figures*) with different languages to meet requirements from different health authorities. For such tasks, SPs normally produce one package in English and then do the translation. It's important to maintain consistency between different packages. There are 2 options for the translation task: one is to translate all the datasets and programs and re-generate another package, the other is to translate the TLFs directly. We prefer the latter one for efficiency and consistency. Sometimes it's the only option when vendor only provides a single combined Docx or RTF file. This paper briefly discusses a SAS® macro tool to automate the translation of RTF files from English to Chinese. Docx files need to be re-saved as RTF files before the translation. This tool can be extended to support translation to other languages. The tool has been integrated into Dizal-iSCP (Dizal-Integrated Statistical Computing Platform), which automates the preparation of TFL packages and translation.

INTRODUCTION

With the increasing number of global clinical trials being carried out in the medical field, translation of clinical trial documents is becoming more and more important. SPs have been using standard programs/macros to generate clinical reports in RTF format for years. The translation of RTF files is not a straightforward project and requires greater precision than translations for text communication in other sectors. In absence of a translation tool, SPs need to prepare and validate an independent package of translated RTF files.

Most of the commonly available machine translation applications like Google Translate do not give precise control of the terminology, which is important to clinical trial documents. Therefore, there is a business need for a tool to automate the translation with precise control of terminology.

The paper introduces our translation process, a SAS® macro tool that utilizes a constantly updated dictionary to replace source language texts with target language texts automatically with customized control of terminology, superscript/subscript, font, timestamp, and so on. The tool extracts all texts from RTF files and hence can also be used to check the consistency of wording.

A limitation of our translation tool is that it does not translate the texts within the figures. SPs need to translate such texts before using the tool. Luckily there are a limited number of such texts, which can be easily handled either manually or automatically.

PREPARATION

Before the translation process, there should be a package of TFLs ready. Besides, there are still possibly 3 steps before the translation process. Firstly, translate the texts within the figures, for the tool cannot translate texts within the figures. Secondly, re-save DOCX files as an RTF files. Thirdly, combine all RTF files to form a single RTF file. It's recommended to begin the translation with a combined RTF file because it will benefit the preparation of a study dictionary. And there will be no need to combine the translated TLF files. An example of a macro tool for RTF combination refers to A Fully Automated Approach to Concatenate RTF outputs and Create TOC.

After these steps, the RTF file can be translated to a report-ready RTF file in another language by using this macro tool.

TRANSLATION PROCESS

This flow chart in Figure 1 illustrates the core of the translation tool. The diagram with bold arrow lines describes how the input RTF files are read into SAS and manipulated step by step, and how the dictionaries are maintained and used for translation.

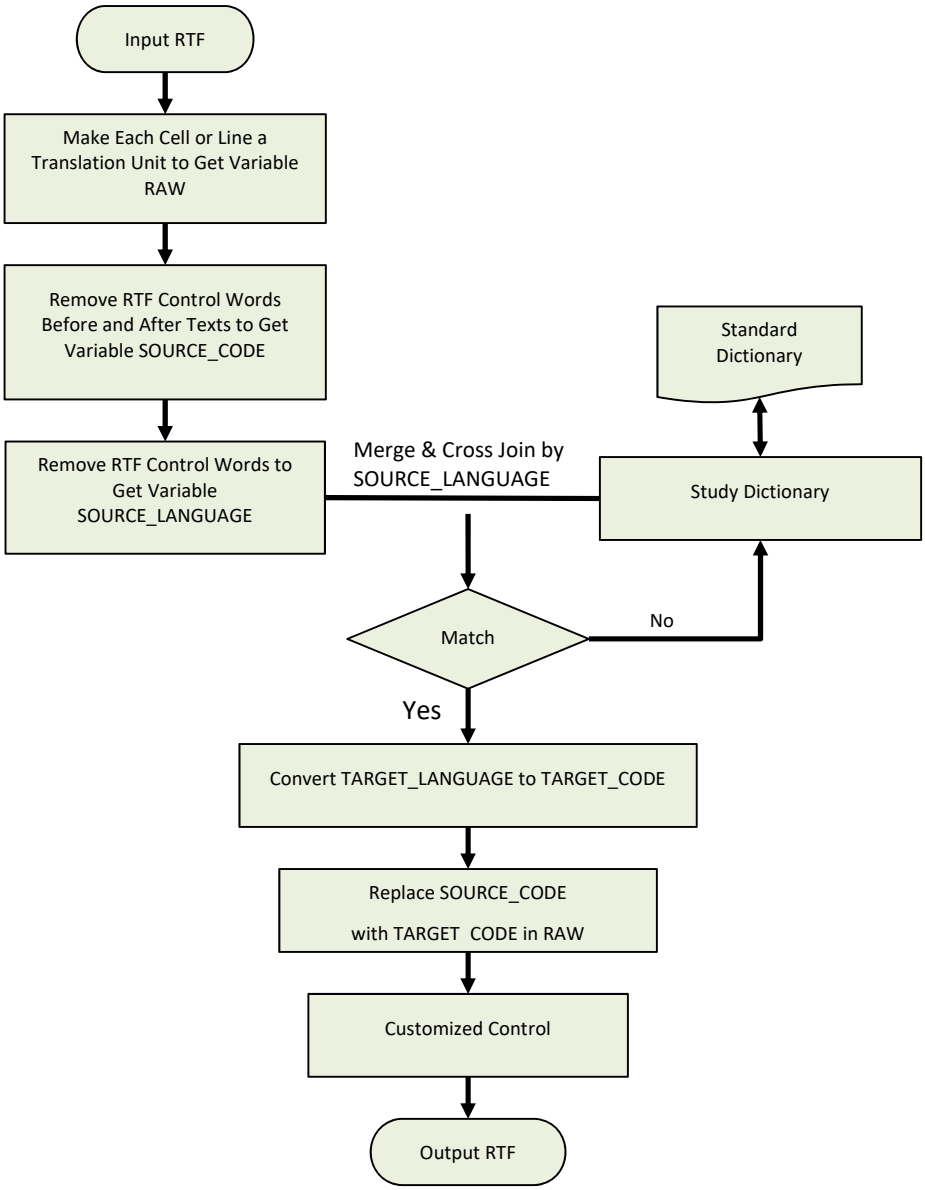


Figure 1. Translation process flow

1. **Sample RTF table**
   Figure 2 shows a sample RTF table in English which will be used as an example to discuss the various steps involved in the table translation process.

PROJECT A Page 1 of 1

Table 1.1.1
Summary of Demographic and Baseline Characteristics
Safety Analysis Set

	ARM A (N = ##)	ARM B (N = ##)	Total (N = ##)
Age			
n	##	##	##
Mean (SD)	##.# (##.##)	##.# (##.##)	##.# (##.##)
Median	##.#	##.#	##.#
Min, Max	##, ##	##, ##	##, ##
Sex, n (%)			
Female	## (##.#)	## (##.#)	## (##.#)
Male	## (##.#)	## (##.#)	## (##.#)
Race, n (%)			
Asian	## (##.#)	## (##.#)	## (##.#)
Ethnicity, n (%)			
Hispanic or Latino	## (##.#)	## (##.#)	## (##.#)
Not Hispanic or Latino	## (##.#)	## (##.#)	## (##.#)
ECOG Performance Status, n (%)			
0	## (##.#)	## (##.#)	## (##.#)
1	## (##.#)	## (##.#)	## (##.#)

Source: ADSL
Data extraction: 14AUG2020, Data cut-off: 07AUG2020
Abbreviation: N=Total number of patients treated; ECOG=Eastern Co-operative Oncology group.
/ar-dev/pgmanal/reports/t-dm.sas 08OCT2020 9:02 t-dm.rtf

Figure 2. Sample RTF table

2. Make Each Cell or Line a Translation Unit to Get Variable RAW

The RTF file is read into a SAS dataset by using a DATA STEP. Then to make every single record a meaningful translation unit, the tool uses the SAS SCAN function with delimiters to separate and combine records to form variable RAW. The delimiters are transferred from RTF control words like \page, \par, \row, \cell, and \line. This step doesn't change the RTF content and structure.
Example SAS Code:

```
%*------------------------------------------------------------------*;
%* Make Each Cell or Line a Translation Unit to Get Variable RAW ;
%*------------------------------------------------------------------*;
data _rtf2;
  set _rtf1;
  if find(rtfcode,'{\field' ) then
  rtfcode=tranwrd(rtfcode,'{\field','~{\field');
  rtfcode=tranwrd(rtfcode,'\sectd ','~\sectd ');
  rtfcode=prxchange('s/\\page |\\page(\W)/\\page~$1/',-1,rtfcode);
  rtfcode=prxchange('s/\\row |\\row(\W)/\\row~$1/',-1,rtfcode);
  rtfcode=prxchange('s/\\cell |\\cell(\W)/\\cell~$1/',-1,rtfcode);
  rtfcode=prxchange('s/\\par |\\par(\W)/\\par~$1/',-1,rtfcode);
  rtfcode=prxchange('s/\\line |\\line(\W)/\\line~$1/',-1,rtfcode);
  rtfcode=prxchange('s/\\tab |\\tab(\W)/\\tab~$1/',-1,rtfcode);
  if ^find(rtfcode,'~') then do;
    rtfcode2=rtfcode;
    ID+1;
    output;
  end;
  else do i=1 to count(rtfcode,'~')+1;
    rtfcode2=scan(rtfcode,i,'~');
    if i>1 then
      ID+1;
    output;
  end;
run;

data _rtf3;
```

```
  set _rtf2;
  by ID;
  length raw $30000;
  retain raw;
  if first.ID then
    raw=rtfcode2;
  else raw=catt(raw,rtfcode2);
  if last.ID;
  len=length(raw);
  keep ID raw;
run;
```

3. **Remove RTF Control Words Before and After Texts to Get Variable SOURCE_CODE**
   The next step is to derive the variable SOURCE_CODE. RTF control words might exist before, after, and within a segment of text. To derive SOURCE_CODE, the RTF control words before and after the text are removed. The value of variable SOURCE_CODE can be found in RAW with SAS TRANWRD function and be easily replaced by TARGET_CODE.
   Example SAS Code:

```
%*-----------------------------------------------------------------*;
%* Remove RTF Control Words Before and After Texts to Get Variable
   SOURCE_CODE;
%*-----------------------------------------------------------------*;
data _rtf4;
  set _rtf3;
  Source_Code=raw;
  Source_Code=prxchange('s/\{\\?\*?\\fldinst
?\{?[^\{\}]+\}+|\{\\pntxt[ab] [^\{\}]+}|\\blipuid \w+|\\datafield
\w+|\{\\sp\{.+\}\}|\{\\s[nv] .*\}|\{\\\*\\\w+ [\w-
]+\}|\\\*|^[\w]{40,}//',-1,Source_Code);
  Source_Code=transtrn(Source_Code,'{\sp}',trimn(''));
  Source_Code=prxchange('s/^[\{\} ]*//',-1,Source_Code);
  Source_Code=prxchange('s/^(\\[a-tv-z][\w-]* ?\{*\}*\{*)*//',-
1,Source_Code);
  Source_Code=prxchange('s/[\{\} ]*(\\[\w-]+ ?\{*\}*\{*)*[\{\}
]*$//',-1,Source_Code);
  Source_Code=left(Source_Code);
run;
```

4. **Remove RTF Control Words to Get Variable SOURCE_LANGUAGE**
   The rest RTF control words which are within a text will be removed in the next step to get variable SOURCE_LANGUAGE. The special characters which are stored as UNICODE are converted to human-readable characters. The only exception is that the RTF control words of superscript and subscript need to be kept in SOURCE_LANGUAGE. Apart from that, SOURCE_LANGUAGE is human-readable and can be used as the key variable to merge and cross join with the study dictionary.
   Example SAS Code:

```
%*-----------------------------------------------------------------*;
%* Remove RTF Control Words to Get Variable SOURCE_LANGUAGE, but
   keep superscript and subscript;
%*-----------------------------------------------------------------*;
data _rtf5;
  set _rtf4;
  Source_Language=Source_Code;
  Source_Language=tranwrd(Source_Language,'\su','#SPECIAL1#su');
  Source_Language=prxchange('s/\\[a-tv-z][A-Za-z]*-?\d* ?|\\u[a-
```

```
z]//',-1,Source_Language);
     Source_Language=prxchange('s/\\u(\d+) ?;?/&#$1;/',-
1,Source_Language);
     if find(Source_Language,'&#') then
       Source_Language=unicode(Source_Language,'ncr');
     Source_Language=left(Source_Language);
     Source_Language=prxchange('s/\r//',-1,Source_Language);
     if find(raw,'\su') then do;
       if substr(Source_Language,length(Source_Language))^='}' then
         Source_Language=cats(Source_Language,'}');
       if count(tranwrd(raw,'\su','#SPECIAL1#su'),'#') ^=
count(Source_Language,'#') then
Source_Language='#SPECIAL1#'!!catx('',scan(substr(raw,find(raw,'\su')),
1,' \'),Source_Language);
Source_Language=prxchange('s/(#SPECIAL1#[^\}]+)\}/$1#SPECIAL2#/',-
1,Source_Language);
     end;
     Source_Language=tranwrd(Source_Language,'\}','#SPECIAL2#');
     Source_Language=tranwrd(Source_Language,'\{','#SPECIAL3#');
     Source_Language=compress(Source_Language,'{}');
     Source_Language=tranwrd(Source_Language,'#SPECIAL1#','{\');
     Source_Language=tranwrd(Source_Language,'#SPECIAL2#','}');
     Source_Language=tranwrd(Source_Language,'#SPECIAL3#','{');
     Source_Language=left(Source_Language);
   run;
```

5. **Merge & Cross Join by SOURCE_LANGUAGE**
   The datasets generated from RTF and dictionary will be firstly merged by SOURCE_LANGUAGE. The records only exist in the dataset from RTF will cross join (cartesian product) with the dictionary to find any records in the dictionary that are part of the records from RTF. If a sentence cannot be found as a whole in the dictionary but can be found as separated segments in the dictionary, it can still be translated segment-by-segment. The merge step is used first because cross join can be very slow and resource-consuming. The merge step can be used to translate texts like AEDECOD, which can be found as a whole in the dictionary (entries for AEDECOD can be downloaded from MedDRA website). If either merge or cross join does not get the corresponding TARGET_LANGUAGE from the dictionary, a not-translated list will be generated and need to be added into the dictionary. SPs need to manually translate them to TARGET_LANGUAGE. By using a null study dictionary, the not-translated list will contain all the texts used in the RTF file and will benefit the wording consistency checking. When the study dictionary is reviewed and updated, SPs need to rerun the tool and check whether all the texts in the source language are translated.

6. **Convert TARGET_LANGUAGE to TARGET_CODE and Replace SOURCE_CODE with TARGET_CODE in RAW**
   TARGET_LANGUAGE normally contains many special characters, which need to be converted to RTF UNICODE by using SAS UNICODEC function. This step is similar to the reverse process of SOURCE_CODE to SOURCE_LANGUAGE.
   Figure 3 as below shows the interim SAS dataset right before the translation for a better explanation of the data flow.

	RAW	SOURCE_CODE	SOURCE_LANGUAGE	TARGET_LANGUAGE	TARGET_CODE
1	\sectd \ltrsect\lndscpsxn\linex0\header...	PROJECT A	PROJECT A	A项目	A\u39033;\u30446;
2	}\pard \ltrpar \qr \li0\ri0\sb10\sa10\sl-23...	Page	Page	页码	\u39029;\u30721;
3	\sectd \ltrsect\lndscpsxn\linex0\endnhe...	of	of	-	-
4	}}{*\pnseclvl1\pnucrm\pnstart1\pninde...	Table 1.1.1	Table 1.1.1	表1.1.1	\u34920;1.1.1
5	\hich\af0\dbch\af31505\loch\f0 Summa...	Summary of Demographic and Ba...	Summary of Demographic and Base...	人口学及基线特征总结	\u20154;\u21475;\u23398;\u21450;...
6	Safety Analysis Set\cell	Safety Analysis Set	Safety Analysis Set	安全性分析集	\u23433;\u20840;\u24615;\u20998;...
7	}\pard \ltrpar\s16\qc \li0\ri0\sb20\sa20\s...	ARM}{\rtlch\fcs1 \af0\afs24 \ltrch\fc...	ARM A	A组	A\u32452;
8	}{\rtlch\fcs1 \af0\afs24 \ltrch\fcs0 \fs20\...	ARM}{\rtlch\fcs1 \af0\afs24 \ltrch\fc...	ARM B	B组	B\u32452;
9	}{\rtlch\fcs1 \af0\afs24 \ltrch\fcs0 \fs20\...	Total	Total	合计	\u21512;\u35745;
10	\ltrrow}\trowd \irow3\irowband3\ltrrow\t...	Age	Age	年龄	\u24180;\u40836;
11	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	n	n	例数	\u20363;\u25968;
12	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Mean (SD)	Mean (SD)	均数(标准差)	\u22343;\u25968;\u65288;\u26631;...
13	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Median	Median	中位数	\u20013;\u20301;\u25968;
14	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Min, Max	Min, Max	最小值,最大值	\u26368;\u23567;\u20540;\u65292;...
15	\ltrrow}\pard\plain \ltrpar\s17\ql \li0\ri0\s...	Sex, n (%)	Sex, n (%)	性别,例数(%)	\u24615;\u21035;\u65292;\u20363;...
16	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Female	Female	女性	\u22899;\u24615;
17	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Male	Male	男性	\u30007;\u24615;
18	\ltrrow}\pard\plain \ltrpar\s17\ql \li0\ri0\s...	Race, n (%)	Race, n (%)	种族,例数(%)	\u31181;\u26063;\u65292;\u20363;...
19	\ltrrow}\pard\plain \ltrpar \s17\ql \li300\ri...	Asian	Asian	亚裔	\u20122;\u35028;
20	\ltrrow}\pard\plain \ltrpar\s17\ql \li0\ri0\s...	Ethnicity, n (%)	Ethnicity, n (%)	种族,例数(%)	\u31181;\u26063;\u65292;\u20363;...
21	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Hispanic or Latino	Hispanic or Latino	西班牙裔或拉丁裔	\u35199;\u29677;\u29273;\u35028;...
22	\ltrrow}\pard\plain \ltrpar\s17\ql \li300\ri...	Not Hispanic or Latino	Not Hispanic or Latino	非西班牙裔或拉丁裔	\u38750;\u35199;\u29677;\u29273;...
23	\ltrrow}\pard\plain \ltrpar\s17\ql \li0\ri0\s...	ECOG Performance Status, n (%)	ECOG Performance Status, n (%)	ECOG状态评分，例数(%)	ECOG\u29366;\u24577;\u35780;\u...

Figure 3. Interim SAS dataset before translation

In this dataset, the RAW variable still keeps the original information from RTF file and will be updated in the next step by replacing the value of SOURCE_CODE in RAW with the value of TARGET_CODE.

7. **Customized control**

   Besides translation, the tool provides additional customized control to format, font, and timestamp. For example, normally there's no blank space between Chinese characters. The tool will automatically remove blank spaces between Chinese characters and replace most punctuation marks with correspondent Chinese characters. It also changes the default font from "Times New Roman" to "Simsun" which is widely used for Chinese characters. There's a timestamp for each output in the footnote. The tool also offers a choice to update the timestamp to the current SYSDATE value. Figure 4 as below shows the sample translated RTF table in Chinese.

A项目 页码 1 – 1

表1.1.1
人口学及基线特征总结
安全性分析集

	A组 (N = ##)	B组 (N = ##)	合计 (N = ##)
年龄			
例数	##	##	##
均数（标准差）	##.#（##.##）	##.#（##.##）	##.#（##.##）
中位数	##.#	##.#	##.#
最小值，最大值	##，##	##，##	##，##
性别，例数（%）			
女性	##（##.#）	##（##.#）	##（##.#）
男性	##（##.#）	##（##.#）	##（##.#）
种族，例数（%）			
亚裔	##（##.#）	##（##.#）	##（##.#）
种族，例数（%）			
西班牙裔或拉丁裔	##（##.#）	##（##.#）	##（##.#）
非西班牙裔或拉丁裔	##（##.#）	##（##.#）	##（##.#）
ECOG状态评分，例数（%）			
0	##（##.#）	##（##.#）	##（##.#）
1	##（##.#）	##（##.#）	##（##.#）

数据来源：ADSL
数据抽取日期：2020-08-14，数据截止日期：2020-08-07
缩写：N=剂量组内的受试者例数；ECOG=东部肿瘤协作组。
/ar-dev/pgmanal/reports/t-dm.sas 2021-03-28 0:10 t-dm.rtf

Figure 4. Sample translated RTF table

MAINTENANCE OF DICTIONARIES

The right side of the flow chart in Figure 1 describes the maintenance of dictionaries. There are 2 kinds of dictionaries with the same structure: standard dictionary and study dictionary. The standard dictionary contains common entries which can be used across studies. The study dictionary only contains entries to be used within the study. At the beginning of the translation process, there's no study dictionary. The standard dictionary is automatically copied to the study folder and used as a study dictionary. After joining with the study RTF file, the tool will remove unnecessary entries from the study dictionary for efficiency. The final version of study dictionaries can be used to update the standard dictionary.

Source Language	Target Language	RTF_NAME
Protocol No.	方案号	
Data cut-off	数据截至日期	
Dry Run	试运行	
Table	表	
Figure	图	
Listing	列表	
Analysis Populations	分析人群	
Part	阶段	
All Patients Screened	所有筛查患者	
Escalation	剂量递增	
Expansion	剂量扩展	
Total	合计	

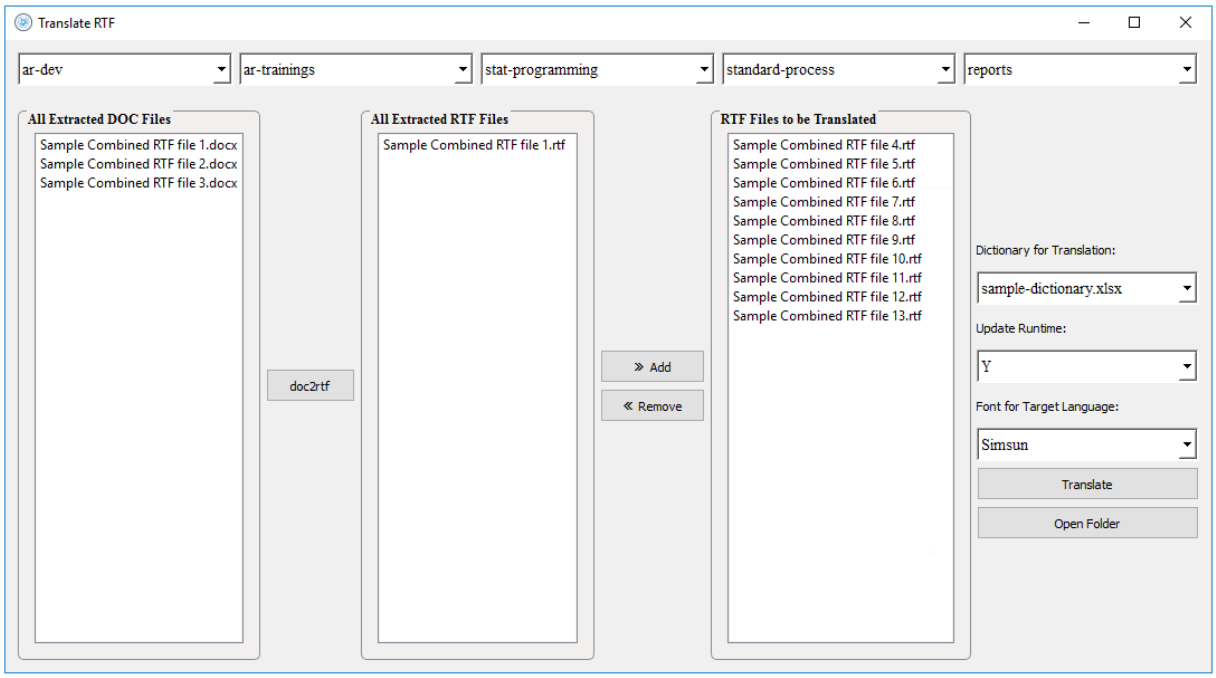
Safety Set	安全性数据集	
PK Set	药代动力学数据集	
SD	标准差	
SD	疾病稳定	t-orr

Table 1. Example of English to Chinese Dictionary

There are 3 columns in the dictionaries: SOURCE_LANGUAGE, TARGET_LANGUAGE, and RTF_NAME. SOURCE_LANGUAGE will be used as a key variable to join RTF files. TARGET_LANGUAGE is translated texts. Both SOURCE_LANGUAGE and TARGET_LANGUAGE are human-readable texts and contain no RTF control word except superscript or subscript. TARGET_LANGUAGE will be transformed to RTF UNICODE by using SAS UNICODEC function to form the variable TARGET_CODE. RTF_NAME is the name of the RTF file which needs special translation. For example, “SD” mostly stands for “Standard Deviation” but can also be interpreted as “Stable Disease”. For such a case, an additional entry of the “SD” should be added with RTF_NAME.

INTEGRATED INTO DIZAL-ISCP

To further improve efficiency and reduce manual work, this translation macro tool has been integrated into Dizal-iSCP as well as other tasks like converting DOCX files into RTF files, combing RTF files, and converting RTF files into PDF files. Dizal-iSCP is an interactive programming interface that was developed by Dizal infrastructure team. It consists of different applications and each application is designed for a specific task. Converting DOCX file into RTF file and calling this translation macro to translate text can be done through application – *Translate RTF*.



After the study path was selected, all existed doc files within the chosen folder will be listed in the left box. While at the same time, all RTF files (if any) will be displayed in the middlebox. By selecting desired DOCX file and clicking on the button ‘doc2rtf’, the DOCX file will be converted into an RTF file which will

be added into the middlebox. Buttons ‘Add’ and ‘Remove’ can help pick out RTF files to be translated. “Add” can help move selected RTF files from the middlebox into the right box. While “Remove” works the other way around. After selecting the correct dictionary file and font, all RTF files in the right box will be translated through the button ‘Translate’.

CONCLUSION

The manual translation is time-consuming and error-prone. This translation tool has greatly increased the quality and efficiency when we implemented it in studies. The standard dictionary collects entries that are commonly used in study dictionaries, which is very important to this tool. With a good standard dictionary, there will be minimal updates to the study dictionary when we work on a new study. The more this tool is used, the more powerful it will be.

REFERENCES

Lugang Xie (Larry), Jundong Ma and Jie Wang (Jerry). “SAS Utility to Combine RTF Outputs and Create a Multi-Level Bookmark Hierarchy and a Hyperlinked TOC”. PharmaSUG China 2019. Available at https://www.lexjansen.com/pharmasug-cn/2019/AD/Pharmasug-China-2019-AD25.pdf

ACKNOWLEDGMENTS

The authors would like to thank Huadan Li, Jianyong Tong, and Jiarui He for their great support and valuable input into this paper.

CONTACT INFORMATION

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

Jundong Ma

Dizal Pharma

E-mail: jundong.ma@dizalpharma.com