Translation: source document quality problems

Gleb Yaltchick, Oleg Vigodsky

Gleb Yaltchick is a software engineer at LONIIS Translation Centre
Oleg Vigodsky is a manager of LONIIS Translation Centre

Keywords

translation, machine translation, translation memory, documentation update, documentation quality, documentation reusability

Summary

Common errors (punctuation, poor and misformatting, undesirable text rewording, etc.) in source documents are generalized and analyzed. The influence of these errors to the documentation translation process is evaluated. Some recommendations to avoid such errors are provided.

Introduction

Different issues, related to writing and publishing technical documentation, are well known for a long time, studied and investigated from different points of view. There are numerous manuals and recommendations which allow to avoid a major part of errors (if not all). However, in almost all these manuals the documentation is considered as a final product, which is not to be processed or updated in the future. Nevertheless, the normal life cycle of the documentation is not limited by writing, editing, making up and publishing only. Actually, the writer has to resolve two further problems, which imply additional requirements to the output of the technical documentation department.

These are:

1. Translation of the documentation

2. The optimal use of currently existing documents when writing and making up the documentation for the new product release.

Though, on the first sight, it seems that these problems are independent to each other, but between them there is a deep interrelation.

On the one hand, it is necessary to prepare the documentation in such a way, that its translation (as an ideal) to any languages would be not very difficult and cause minimum sense loss and distortion, as well as minimum loss of integrity.

On the other hand, it would be necessary to achieve the case that if we use the source documentation of a previous version when preparing its new release, then while translating the new version it would be possible to use the previous release of the document translated earlier.

One should think, there are no obstacles, and if the author has created the new version on the basis of the old one, then the translator who has the old version of a source document and its translation, as well as the new version, would generate (without specific costs) a new translation of the document. Actually, it is not so. Why it is not so, which problems arise and which methods to avoid these problems we see - all these issues are covered by this article.

Below we consider this issue from the point of view of people, involved to the documentation translation process, and first of all,the translation of the new release of the document translated previously. We do not analyze a documentation writing process, because it is well done by others.

Why it is not so

In fact, there are many problems. All of them are whether technical or psychological.

Psychological problems are very simple. Nobody can (physically) fully use all the information, which is contained in the documentation already translated. It is impossible to remember all text already translated, or to search for changed fragments manually by comparing two document versions.

It is obvious, that a maximum reachable is the repetition of translation style of the previous document. It is not too bad, if to take into account, that usually in

translation process several people are involved, each translator translating his/her own part, and frequently new versions of document are passed to other translators.

Actually, at present, nobody performs such a work only manually. However, available tools unfortunately are not perfect and do not resolve all problems.

One, Two, Three...

There are some methods to reduce own work by using old versions.

- The first method is elementary. While preparing a new version try to handle a minimum number of separate documents, do not change a text structure and order the translation of only those files, which were changed.
- The second method is to take changed documents, to find a text, which was not changed and to translate new/modified text.
- The third method is to analyze the previous version of documentation and its translation, and then on the basis of this analysis, to translate automatically all possible analogues in the new version of the documentation. All further activities are up to translators.

Now we will consider more in detail each of these three methods.

Not all simple is ingenious

The most important is as follows. It is not always possible to find even one file, which was not changed. Moreover, frequently a documentation structure is modified, and causes significant changes in document organization, for example, when changing a medium (generation of on-line documents in HTML format).

Note that even those documents, which "were not changed", will cause some extra work, but it is already extremely technical processing such as updating page references and changing version number in header/footer.

On practice, this method works poorly.

Let persuer procure

It sounds well, "to find everything that was not changed, and to translate all other text". Unfortunately, the situation is not so good.

One of main problems is „**to find**“.

A human easily can say, whether the text was changed or not. It is impossible to define automatically (in every case). The computer can resolve this problem (practically with no errors) only when it compares the "raw" text.

It starts comparing a formatted text (it is more difficult), so the quality of such comparison is reduced significantly. Sometimes, the result is unacceptable.

The reason is that to compare such a text, the computer should have the detailed information on an internal text structure. As a rule, current documentation preparation tools have built-in document comparison features, however even in a case, when such features are realized at a decent level (for example, Frame products), it can not be guaranteed, that the comparison will be satisfactory. By

contrast to the human, modern DTP tools do not support "fuzzy logic" and "multi-factor analysis" features, so the comparison is performed at a "exact coincidence" level. Due to simple style changes or picture inserts, the comparison feature could not identify absolutely identical text fragments, although, from the user's point of view they are the same.

The second problem is a modification of document logic. If you change some paragraphs (their places) in the document, the comparison feature will decide that all the document was changed (although, in such a case it is enough to rearrange some paragraphs in old translation).

It is acceptable method. Many people even use it. On the one hand it is simple and implies low technical requirements. On the other hand, the quality is not too high.

No one can brace unbounded or from difficult to simple

The third method looks as the most attractive one. Unfortunately, at the moment, it is not fully impracticable. There are no general purpose software packages, which would permit to conduct such analysis. Alas...

However, in practice, we do not need to analyze the whole text. It is enough to be limited by sentences (as an atomic item) to be analyzed. Thus we do not need to pay attention to a sentence contents. It is enough to define that such a sentence is an independent, constant "brick", always having a unique translation. Such "bricks" (or segments) shall be used while translating a new document.

Such working method was called a „Translation Memory".

The process of text processing is as follows:

- Split the previous source document version to "bricks" (segments),
- Split its target translation to the same number of segments and to arrange pairs (source segment to appropriate translation segment),
- Split the new source document version, using the same approach,
- Check for each source segment, whether there is a precisely same target (translated) segment in translated document, and if yes, to replace it with translation segment.

Certainly, this it slightly simplified model (but slightly only). There is always a small probability to make an error due to the fact, that in a given context the sentence will have a different translation. But as an experience shows, such errors are extremely rare, and if a documentation is well prepared, it is possible to get up to 75-80 % of "pretranslated" text.

Now, after having introduced various approaches, we will see some obstacles.

The patient is whether live or dead

We can say about the first method few words only. The file is whether changed, or not. It is impossible to do anything here.

Remaining activities are done under administrative methods only and are not disclosed in this article.

Do not repair anything that was not broken

The comparison of two documents within a DTP tool is the more difficult task, and here is a number of recommendations (quite obvious ones).

The most important desire is as follows: do not change anything if you do not need.

- If there is no real necessity to change anything within a paragraph, please leave like it is.

```
If there is no real necessity to change anything
within a paragraph, please leave like it is.

If you do not really need to change anything within a
paragraph, please leave like it is.
```

Example 1: Don‘t make changes without necessity.

- Do not change a template of a whole document.

Usually, changing a template is a simple task. For you. Probably, after this operation the document will look even more attractively, than previously. However, please think about those people, who will then be involved in translation process. The comparison feature will identify the text as completely changed one, and if the translation process is arranged as a flow, than very likely the translator will not evaluate, what is the reason and simply translate this text once again. As a customer, you will spent extra time and money.

- Try not to change a paragraph layout.

If you need to modify a paragraph, then correct its contents only.

- If you create a new version on the basis of the old version and decided to emphasize something using e.g. underlining or BOLD font, please follow the previous document style.

The ideal variant is to save a previous style in your "new" document, but not vice versa. Your new version can even look much better, but remember, that if the new document is to be translated to 5 foreign languages, you will add extra work to at least five people more.

- If you insert a cross-reference marker, or add other "invisible" information to the document, please insert it into the end or in a beginning of the paragraph. If you will insert it into a middle of a text, then the paragraph will be marked as changed (though the marker inserted is invisible, but for a publishing tool it exists).

Wrong:

```
Please insert "invisible" mar⊥kers into the end or in
the beginning of the paragraph
```

Right:

```
⊥Please insert "invisible" markers into the end or in
the beginning of the paragraph
```

Example 2: Marker location

- If possible, do not rearrange text parts, paragraphs or even the chapters.

Sweet for the dessert

The third method is that with document analysis or, in other words, using Translation Memory.

This is a method, being the most perspective, which gives frequently excellent results, where anybody does not expect it, though the initial document is changed significantly. On the other hand, it provides completely unacceptable results with the documents, which visually are practically the same.

Ironically, all those recommendations, mentioned for a document comparison method, do not have any serious influence to the third method.

Let us remember these recommendations:

- Do not change a document template

The modification of a template does not influence the text contents of the document, and the segmentation process either.

- Do not change the paragraph layout

All the information on paragraph format is excluded while creating a segment. Therefore two sentences, which have an identical contents, but different styles, will be the same segments.

- Do not change a paragraph design

The information on such changes is specially coded, and though it is saved within resulting segments, nevertheless the program of searching paired segments can ignore such a code. As a result, even if you emphasize a couple of words by underlining, then resulting segments will visually differ from each other, but

during "pretranslation" all such segments (distinguished by the service information only) will be identified as identical.

All above mentioned relates to both marker positions and hidden information (within text paragraphs).

- Do not change paragraph positions

It is clear, that as we compare each segment separately with reference material (the base of previously translated segments), while the order of comparing such segments does not matter.

We mentioned what does not interfere normal application of this method. Now we will evaluate, what can interfere.

For better understanding a further explanation, we would describe a segmentation process in detail.

- all the information on a document layout and formats is removed
- all changes in text formatting within paragraphs, hidden service information such as markers and the similar items are excluded from the text, however instead a special marker is inserted into the text (which is ignored by the segment comparison program, but enables subsequently to restore all formatting)
- a text is divided into separate sentences using several attributes:
  - dot,
  - question mark,
  - exclamation mark at the end of the sentence
  - end of paragraph
  - end of page
  - end of chapter
- checking exceptions (dots in acronyms such as "etc.", special characters in file names, acronyms ("FRAME.A") and so forth)

Each resulting "sentence" is considered as a separate segment.

Source text:
The ***EWSD*** system provides different features, e.g. ***call fowarding***. DLU-B unit terminates subscriber lines. These lines can be:¶
1) → digital,¶
2) → ISDN,¶
3) → analog.¶

Resulting segments:
```
[1]   The |EWSD| system provides different features,
      e.g. |call fowarding|.
[2]   DLU-B unit terminates subscriber lines.
[3]   These lines can be:
[4]   digital,
[5]   ISDN,
[6]   analog.
```

Example 3: Segmentation.
In the example above, symbol ¶ means "end of paragraph", → is a tabulation character. For the listing, autonumbering style is used. Note that 1), 2) and 3) are generated by an autonumbering feature and are not included in result segments. Any changes of style within a sentence (***EWSD, call forwarding***) are marked using a special character (|) in resulting segment.

Two segments are considered as identical, if and only if they exactly coincide with each other (except for service markers and, as a rule, digits).

`Identical`:
```
These are identical segments created in 1998
These are |identical| segments created in 1999
```

`Different`:
```
E.g., these segments will be different
E.g. these segments will be different
```

Example 4: Identical and different segments

So, by breaking the document and by comparing segments with existing translated segment base, we will get whether well, or poorly pretranslated document.

How to get a well pretranslated document?

1. First of all, let us exclude that parts of text, which generally are not to be translated. These are program listings, menu items, key names, shortcuts and so on.

If, for example, file "XXX.CFG" is selected, a screen with following information appears:

```
C:\TEST\XXX.CFG   DESTINATION:      XXXXX
                  SERIAL NUMBER:    XXXXX
                  OPERATOR NAME:    XXXXX
```

Example 5: text part (screen) not to be translated

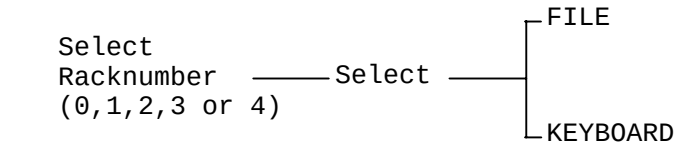
It is obvious, that best way (for automatic translation) is not to process it at all, but to mark it as translated and to put it to the target document.

The translation of such text is a superfluous work, but there is one more reason to exclude such a text from consideration.

As a rule, these are whether short lines with one to two words, that provides much higher probability of wrong translation, than for the usual sentence, or this is a text with a lot of special characters and digits (for example, program listings).

The quality of segmenting such a text is low (accurate segmentation process for this case is to be very complex because of unpredictable text structure).

Source document fragment



Resulting segments:

```
Select|Select|FILE
Racknumber
(0,1,2,3 or 4)
KEYBOARD
```

Example 6: Segmentation of "short" text lines

Our work is simplified, if for such paragraphs in the document a special style is used. For example, you can create a special style "listing" and to apply it to listings only.

- Use for a text, not to be translated, a special layout within a documentation format (special paragraph style, special character style, unique combination of fonts and sizes, special elements of a document structure, or any other method, which will allow to uniquely identify such a text)

2. The second problem is tables

The most awful case which we identified is as follows. We got a document, where the tables were made as text lines, in which column gaps were made using spaces, and over a text graphic lines were drawn (using a built-in editor of a publishing tool). No doubts, we were forced to translate all the text in such "tables" manually.

Fortunately, such cases are rare, however, even when the tables are created correctly, results are not so good.

The experience shows, that the first reason is writers frequently divide the logically uniform sentence to some paragraphs to achieve a more beautiful document layout. In that case, one sentence will be splitted to a number of small segments, which are practically not suitable for automatic translation.

This problem can be easily resolved if to use a line break symbol rather than splitting to paragraphs. Thus it is possible to get both an accurate text (in a cell) and a sentence, not broken to different paragraphs.

Source document fragment with two paragraphs

Reporting Level	**Comments**
Subrack	Can usually be cleared with¶ the reset button RT

Resulting segments:

```
[1] Reporting Level
[2] Comments
[3] Subrack
[4] Can usually be cleared with
[5] the reset button RT
```

Source document fragment with paragraph break

Reporting Level	**Comments**
Subrack	Can usually be cleared with↓ the reset button RT

Resulting segments:

```
[1] Reporting Level
[2] Comments
[3] Subrack
[4] Can usually be cleared with|the reset button RT
```

Example 7: Segmentation of sentences within table cells

Besides, there are tables, where one or more columns have no information, to be translated. Considering above explanation, the best solution is to use (for such columns) a special style.

- Do not split a sentence in one cell to different paragraphs,

- Emphasize columns (using a special style) not to be translated

3. It happens, when creating a new version of the documentation using the previous one, that the writer changes styling of sentences, rewriting existing text, but not modifying its meaning.

Such situation usually occurs, when the writer adds one or more new sentences, and then changes the style or grammar of the old text according to new one. As a result, one should translate both the new text, and the old one (although its meaning was not changed).

An example is the global replacement of any text fragment, e.g.,
`Unit 'A'` to
`Unit-A`.

Sometimes, it is really needed but, as a rule, the reason is the aesthetic predilection of the writer only.

- When adding new text to existing one do not change an old information.

4. The next problem is a different spelling of terms, acronyms and abbreviations by various writers or even by the same writer in one document.

For example,

`Unit (B)`	and	**`Unit(B)`**,
`e.g.`	and	**`eg.`**
`(c)`	and	©

and so on.

- Try to avoid such a case (we can not recommend nothing more). Use a autotext in word processors, which support such a feature. Remove this at proof-reading stage. Create a shared glossary of terms and acronyms.

5. Autonumbering and creating lists in documents.

The majority of modern DTP packages support an autonumbering of paragraphs or emphasizing lists and numbered lists using special characters.

However, it can result in significant problems while preparing the new version of documentation, if in the previous version such lists were made manually.

When formatting lists manually, characters for emphasizing a start of list items or the autonumbering are an integral text part. When formatting these automatically, such lists are a part of a control information and, as result, are not available in extracted text segments.

It is obvious, that absolutely identical (on a hard copy) sentences will be identified by a translation memory tool as different text segments.

- If the source document does not use the autonumbering or similar styles with automatic text generation, do not use autogeneration, if possible.

6. Such simple element as quotes can become a stumbling-block as well.

The conventional fonts contain different types of quotes, most frequently used among which are ASCII, opening and closing ones. All these quotes have different designations and codes, which differ from each other (from the point of view of the translation memory tool).

The ASCII code contains only ASCII quotes, which are entered at the keyboard directly (without using specific programs). Many DTP packages, on the contrary, while typing a text, automatically convert them to opening/closing quotes. In the worst case, this is done without any intervention of the user.

As a result, after correcting the text, there is a possibility to create a document, where all ASCII quotes, used in the previous version, are replaced by curved ones, so the percent of reusing is minimized.

ASCII quotes
This is an example of using "autocorrect" feature

Word‘s "smart" quotes
This is an example of using “autocorrect” feature

Word’s "autocorrect" quotes
This is an example of using „autocorrect“ feature

Example 8: Autocorrected quotes

- Do not use built-in DTP features for autocorrecting quotes. The simplest way is to use ASCII quotes only, or if such a case is unacceptable, to replace all types of quotes in all documents to a single type. Then, when translating, it is possible to perform a similar operation for the source document and to get matching segments.

Conclusion

Conformance to above recommendations and wishes will allow to:

- minimize time costs during translation,
- improve consistency of translated documents,
- minimize a translation price,
- make a translation process more comfortable for translators.