Creation of new TM segments: Fulfilling translators' wishes

Carla Parra Escartín
Hermes Traducciones
C/ Cólquide 6, portal 2, 3.ž I
28230 Las Rozas, Madrid, Spain
carla.parra@hermestrans.com

Abstract

This paper aims at bridging the gap between academic research on Translation Memories (TMs) and the actual needs and wishes of translators. Starting from an internal survey in a translation company, we analyse what translators wished translation memories offered them. We are currently implementing one of their suggestions to assess its feasibility and whether or not it retrieves more TM matches as originally expected. We report on how the suggestion is being implemented, and the results obtained in a pilot study.

1 Introduction

Professional translators use translation memories on a daily basis. In fact, most translation projects nowadays require the usage of Computer Assisted Translation (CAT) tools. Sometimes translators will freely choose to work with a particular tool, and sometimes it will be the client who imposes the usage of such tool. One of the main advantages of CAT tools is that they allow for the integration of different productivity enhancement features.

Translation Memories (TMs) are used to retrieve past translations and partial past translations (fuzzy matches). Terminology databases (TBs) are used to enhance the coherence on terminology and to ensure that the right terminology is used in all projects. Moreover, some CAT tools offer additional functionalities such as the usage of predictive texts ("Autosuggest" in SDL Trados Studio[1] and "Muses" in MemoQ[2]), the automatic assembly of fragments to produce translations of new segments, or specific, customizable, Quality Assurance (QA) features.

[1] www.sdl.com
[2] www.memoq.com

In the context of a translation company, the usage of these productivity enhacement tools is part of the whole translation workflow. Project managers use them to generate word counts and estimate the time and resources needed to make the translation. They also use them to pre-translate files and clean them up prior to delivery to the client. Translators use these tools to translate, revise and proofread the translations prior to final delivery.

Several researchers have worked on enhancing TMs using Natural Language Processing (NLP) techniques (Hodász and Pohl, 2005; Pekar and Mitkov, 2007; Mitkov and Corpas, 2008; Simard and Fujita, 2012; Gupta and Orăsan, 2014; Vanallemeersch and Vandeghinste, 2015; Gupta et al., 2015). Despite reporting positive results, it seems that the gap between academic research and industry still exists. We carried out an internal survey in our company to detect potential new features for TMs that could be implemented by our R&D team. The aim was to identify potential new features that project managers and translators wished for and that would enhance their productivity. In this paper, we report on the results of that survey and further explain how we are implementing one of the suggestions we received. The remainder of this paper is structured as follows: Section 3 summarises the survey we carried out and the replies we got and our company is briefly introduced in Section 2. Section 4 explains how we implemented one of the suggestions we received. Subsections 4.1–4.3 describe how we created new TM segments out of existing TMs. Section 5 reports on the evaluation on a pilot test to assess the real usability of this approach. Finally, section 6 summarises our work and discusses future work.

2 Our company

Hermes is a leader company in the translation industry in Spain. It was founded in 1991 and has

Proceedings of the Workshop on Natural Language Processing for Translation Memories (NLP4TM), pages 1–8,
Hissar, Bulgaria, Sept 2015.

a vast experience in multilingual localisation and translation projects. With offices in Madrid and Málaga, the company has broad knowledge and experience in computer-assisted translation software and specific localisation software, including SDL Trados, memoQ, Déjà Vu, IBM TranslationManager, Star Transit, WordFast, Catalyst, Passolo, Across, Idiom World Server, Microsoft Helium and Microsoft Localisation Studio, among others.

Our company objectives are built upon a solid foundation which have allowed us to achieve a double quality certification for translation services (UNE-EN-15038:2006 and ISO-9001:2008 standards). Our in-house translators and project managers commit daily to provide our clients with high quality translations for different specialised fields (IT, medicine, technical manuals, general texts, etc.).

3 Internal survey

We asked our in-house translators and project managers for potential new functionalities that CAT tools could offer them. More concretely, we asked them which was, according to them, the "missing functionality" as far as TMs are concerned. In total, 10 project manager and 14 translators participated in this internal survey. While not all had a clear idea of what could be implemented, some interesting suggestions came up.

We gathered ideas such as scheduling automatic TM reorganisation to prevent that large TMs end up corrupted. It is a well known fact that large TMs need to be periodically reorganized (i.e. reindexed and eventually cleaned-up). While this feature is available in standard CAT tools such as Studio 2014[3] and memoQ[4], it is not always carried out automatically and it is difficult to estimate when such reorganisation should be carried out. Scheduling it to be run automatically when a particular number of segments (e.g. 500) have been added since the last reorganisation, for instance, would prevent the loss of a TM because of bad maintenance.

Ideas more related to NLP included the automatic correction of orthotypography in the target language, and allowing for multilingual TMs where several source and target languages can be used for concordance searches at once. Although currently it is possible to use multilingual TMs in CAT tools, when starting a new project a language pair has to be selected. This leads to an underusage of the TM, and the potential benefits of querying multiple languages at once are missed. If, for instance, the TM contains a translation unit (TU) for a different pair of languages (e.g. German > Italian) than the ones selected for that specific project (e.g. German > French) and the same source sentence is appearing in the text currently being translated, the translation into a different target language will not be shown (i.e. the Italian translation of the German TU will not be matched). The same would occur with concordance searches. While matches for a different language pair will not be used in the translation, they may be useful for carrying it out. If a translator understands other target languages, these translations may give them a hint as to how to translate the same sentence into their mother tongue[5].

Finally, an interesting idea was to generate new segments on the fly from fragments of previously translated segments. Flanagan (2015) offers an interesting overview about the techniques used by different CAT tools for subsegment matching. Here, we will focus on memoQ's such functionality: "fragment assembly"[6].

Figure 1 shows how this functionality works in memoQ. As may be observed, memoQ looks

[3]In previous versions of Studio, TM reorganisation was available for all types of TMs in the "Translation Memory Settings Dialog Box > Performance and Tuning". As of Studio 2014, file-based TMs are automatically reorganised, while server-based TMs require periodical reorganisation. For further information, see: `http://producthelp.sdl.com/SDL\%20Trados\%20Studio/client_en/Ref/O-T/TM_Settings/Translation_Memory_Settings_Dialog_Box__Performance_and_Tuning.htm`

[4]memoQ actually has a repairing function, the "Translation memory repair wizard", which aims at repairing (i.e. re-indexing) a corrupted TM. According to the documentation, it is also possible to run this function on TMs which are not corrupted. For further information, see: `http://kilgray.com/memoq/2015/help-en/index.html?repair_resource3.html`

[5]For SDL Studio there seems to be an external app, AnyTM, that allows users to use TMs having different language pairs than the ones in the current translation project. As of Studio 2015, this app has been integrated in the CAT tool and become a new feature. However, this tool does not seem to support the usage of truly multilingual TMs (TMs including several target languages for each segment). For further information on the tool, see: `http://www.translationzone.com/openexchange/app/sdltradosstudioanytmtranslationprovider-669.html`).

[6]For further information, see: `http://kilgray.com/memoq/2015/help-en/index.html?fragment_assembly.html`.

for fragments of the source sentence in other TM segments and internally computes their alignment probabilities. It then inserts the translations into the source segment and suggests this new, sometimes partially translated sentence, as a match. Alternatively, only the fragments translated will be inserted in the target segment, one after another, without the source sentence words that could not be retrieved.

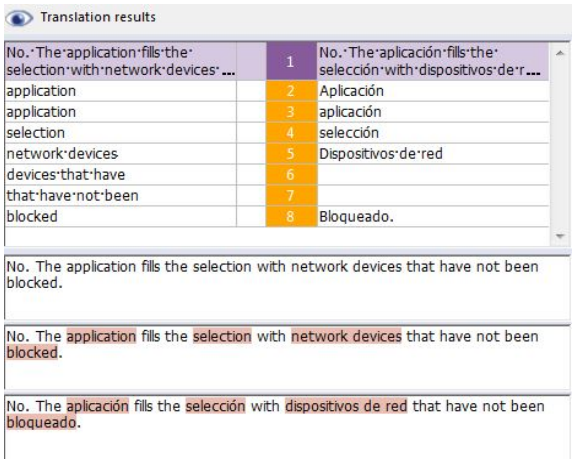

Figure 1: memoQ's fragment assembly functionality.

One limitation of this functionality is that the fragment translations follow the order in which they appear in the source language. Thus, while it may be very useful for pairs of languages which follow a similar structure, it may be problematic for pairs of languages which require reordering. memoQ uses the frequencies of apparition of the fragments to select one translation or another for each particular segment. As a consequence, in some cases the translation selection is wrong, thus yielding wrong translation suggestions.

Examples 1 and 2 show two similar sentences in our TM.

(1) EN: The following message then appears: "Click accept to run the program".
ES: Aparece el siguiente mensaje: "Haga click en aceptar para ejecutar el programa".

(2) EN: The window will show the following: "The application will close".
ES: La ventana mostrará lo siguiente: "Se cerrará la aplicación".

Now imagine we need to translate the sentence in 3.

(3) EN: The following message then appears: "The application will close".

Taking the part of 1 before the colon and the part of 2 after the colon, we would be able to produce the right translation, as shown in 4.

(4) ES: Aparece el siguiente mensaje: "Se cerrará la aplicación".

In technical texts it is often the case that situations like the one just described happen more than once. Thus, it is not surprising that the translators liked the idea and thought it would be nice to find a way of automatically retrieving their translations without having to do concordance searches in the TM. Moreover, remembering that a particular fragment of a segment had been translated in the past is not always possible, as translators may have forgotten it, or different translators may have been involved in the project, thus not seeing fragments of a segment that other translators have translated already.

As this idea seemed to have a great potential to increase the number of TM and fuzzy matches, we decided to implement it and test whether it actually worked. The next Section (4) explains how we proceeded.

4 The new segment generator

As explained in Section 3, we decided to test one idea originated from our internal survey. The idea was to generate new TM segments from fragments of already existing segments. We called our new tool "new segment generator".

The first step was to assess the type of texts that are translated in our company and identify the segment fragments that could be easily extracted. Upon analysis of several sample texts we identified 7 different types of fragments we could work with:

1. Ellipsis
2. Colons
3. Parenthesis
4. Square brackets
5. Curly braces
6. Quotations
7. Text surrounded by tags

In the following subsections (4.1 – 4.3) we describe how each of these types of fragments was treated.

4.1 Ellipsis and colons

One possible type of segment would be that in which an ellipsis ("...") or a colon (":") is used in the middle of the segment. In software localisation or user guides sentences such as the ones in 5 and 6 could appear.

(5) EN: Installing new services... Service XXXX for premium clients: [2]
ES: Instalando nuevos servicios... Servicio XXXX para clientes premium: [2]

(6) EN: You can use the line [abcdef] to describe any of the following characters: a, b, c, d, e, f.
ES: Puede utilizar la línea [abcdef] para describir cualquiera de los siguientes caracteres: a, b, c, d, e, f.

If a different segment only including the text before the ellipsis appears as in Example 7, the TM may not retrieve any fuzzy match. The same would occur with other sentences with colons in which the fragment before or after the colon appears.

(7) Installing new services...

In these cases, we proceded as follows:

1. Check that there is an ellipsis / a colon on both the source and the target segment.
2. Split the segment in two, being the first part the fragment of the segment up to the ellipsis/colon and the second part the fragment of the segment after the ellipsis/colon.
3. Create a new TM segment for each fragment.

4.2 Parenthesis, square brackets and curly braces

Sometimes, a sentence includes a fragment between parethesis, square brackets or curly braces. The content within such characters may constitute a new segment on its own or appear in a different sentence. At the same time, it may also be the case that the same sentence appears in the text, but without such parenthesis. When sentences like the ones in Examples 8–10 appear, it may thus be desirable to store the translation of the fragment between the aforementioned characters and the sentence without such content.

(8) EN: Creates an installation package for application installation (if it was not created earlier).
ES: Crea un paquete de instalación para la instalación de la aplicación (si no se creó antes).

(9) EN: <return code 1>=[<description>]
ES: <código de retorno 1>=[<descripción>]

(10) EN: Could not open key: [2]. {{ System error [3].}} Verify that you have sufficient access to that key, or contact your support personnel.
ES: Error al abrir la clave: [2]. {{ Error en el sistema [3].}} Compruebe que dispone de los derechos de acceso

The strategy to create new segments was the following:

1. Check that there is content between parenthesis / square brackets / curly braces on both the source and the target segment.
2. Keep three fragments out of each sentence:
   (a) A sentence removing those characters and the content between them.
   (b) A fragment starting at the opening character and finishing on the closing one and including the content within. In this fragment, the parenthesis, square brackets or curly braces are mantained.
   (c) A fragment containing only the content withing those characters (parenthesis, square brackets or curly braces), but without them.
3. Create a new TM segment for each fragment.

At this preliminary stage, we considered that when a sentence has several clauses in parenthesis, square brackets or curly braces, these appear in the same order in the target language. This was done so because for the type of texts used so far to test our application (software manuals) and the pair of languages used (English into Spanish), this seems to be the usual case. In future work, we plan to further evaluate this issue, and consider other ways of ensuring that the right translation is assigned to each clause.

4.3 Quotations and text within tags

Quotations and double tags appearing in the text were handled differently. As the text within the quotations or tags might be part of the sentence where it appears, it could not be removed without adding too much noise to the data. Thus, we identified sentences with quotations and/or tags, we then removed the quotations and/or tags and kept the same sentence without them as a new segment. Finally, we also kept the text within the quotation marks or tags as new segments. Examples 11–12 illustrate this kind of segments.

(11) EN: "You can only set the values of settings that the policy allows to be modified, that is, ""unlocked"" settings."
ES: "Solo se pueden establecer los valores de los parámetros que al directiva permite modificar, es decir, los parámetros ""desbloqueados""."

(12) EN: If you clear the <1>Inherit settings from parent policy</1> check box in the <2>Settings inheritance</2> section of the <3>General</3> section in the properties window of an inherited policy, the ""lock"" is lifted for that policy.
ES: Si anula la selección de la casilla <1>Heredar configuración de la directiva primaria</1> en la sección <2>Herencia de configuración</2> que aparece en la sección <3>General</3> de la ventana de propiedades de una directiva heredada, se abrirá el candado para esa directiva.

5 Pilot test

Before integrating our system in a CAT tool and in our normal production workflows, we deemed it better to run a pilot test. The aim of this test was to measure to which extent the new segments retrieved an increased number of 100% and fuzzy matches.

5.1 Test set

We used as a test set a real translation project coming from one of our clients. It is a software manual written in English and to be translated into Spanish. We selected memoQ 2015 to be the CAT tool used for our testing because it is one of the common CAT tools used by our translators and because we also wanted to measure the impact of our approach when using its "fragment assembly" functionality.

The project had in total 425 segments accounting for 6280 words according to memoQ. Table 1 shows the project statistics as provided by memoQ's analysis tool using the project TM provided by the client. Additionally, memoQ identified 36 segments (418 words) which could be translated benefiting from its "fragment assembly" functionality, which uses fragments of segments to create new translations.

TM match	Words	Segments
Repetitions	1064	80
100%	0	0
95-99%	4	2
85-94%	0	0
75-84%	285	14
50%-74%	2523	187
No Match	2404	142
Total	**6280**	**425**

Table 1: Project statistics according to memoQ using the project TM.

Taking this analysis as the starting point of our pilot test, we generated new segments using the approach described in Section 4. We used three different TMs to further assess whether the size of the translation memory matters for generating translations of segments and retrieving more translations. The first TM (Project TM) was the project TM provided by the client. The second TM (Product TM) was a TM including all projects done for the same product of the client. Finally, the third TM (Client TM) included all projects of that client and thus was the biggest one. Table 2 summarises the size of the three TMs.

	Segments	Words		
		EN	ES	
Project TM	16,842	212,472	244,159	456,631
Product TM	20,923	274,542	317,797	592,339
Client TM	256,099	3,427,861	3,951,732	7,379,593

Table 2: Size of the different TMs used.

We then generated new TM segments and stored them as new TMs. Table 3 shows the number of new segments generated using our approach.

Table 4 breaks down the number of segments generated using each strategy and for each TM

	Segments	EN	ES
Project TM new segments	6,776	56,973	66,297
		123,270	
Product TM new segments	7,760	71,034	83,125
		154,159	
new Client TM new segments	74,041	662,714	769,705
		1,432,419	

Table 3: Size of the new TMs generated using our approach.

used. As can be observed, some types are more productive than others. When using the smaller TMs (project and product), the most prolific segment generator category was the one which extracted text surrounded by tags. However, when using the whole client's TM, the text between parenthesis was more prolific.

	TM Proj.	TM Prod.	TM client
Ellipsis	7	6	50
Colon	1801	2094	17894
Parenthesis	1361	1621	**29637**
Square bra.	78	73	598
Curly braces	0	0	0
Quotations	1085	1146	8797
Tags	**2523**	**2892**	17792

Table 4: Number of newly generated segments per type and TM used.

We then tested how many segments would be retrieved using our newly created TMs, both alone and in combination with the TMs we previously had. MemoQ offers the possibility of activating and deactivating different TMs when preparing a file for translation. We thus prepared the project file using 11 combinations to assess which combination performed better as well as whether the new TMs where having any impact in the project. These 11 scenarios were the following:

1. **TM1**: The project TM as provided by the client.
2. **TM2**: Only the new segments generated from the project TM provided by the client.
3. **TM3**: A combination of the project TM and the new segments retrieved from it.
4. **TM4**: Only the new segments generated from the product TM.
5. **TM5**: The project TM and the new segments generated from the product TM.
6. **TM6**: The project TM combined with the new segments TMs generated from the project TM and the ones from the product TM.
7. **TM7**: Only the new segments generated from the client TM.
8. **TM8**: The project TM combined with the new segments generated from the client TM.
9. **TM9**: The project TM combined with the new TMs generated from the client TM and the ones from the project TM.
10. **TM10**: The project TM combined with the new segments generated from the client TM and the ones from the product TM.
11. **TM11**: The project TM combined with the new segments generated from the client TM, the ones from the project TM and the ones from the product TM.

The preparation of a file for translation typically includes both analysing the file and pretranslating it. When the fragment assembly functionality from memoQ is activated, information about how many segments could be translated using fragments is also provided. Tables 5 and 6 summarise the results we obtained for each TM environment when preparing the project for translation.

As can be observed, using the TMs with new segments decreased in all cases the number of segments not started and increased the number of segments translated using fragments (cf. Table 5). It also seems clear that size matters and that the bigger the TM with new fragments, the higher the number of segments that benefit from fragments (cf. TM2, TM4 and TM7 in Table 5).

However, this does not hold true for the pretranslation. In all cases in which the new TMs were used in isolation (TM2, TM4 and TM7) the number of pretranslated segments decreases. This may be due to the fact that those TMs only contain fragments of the original segments present in the different TMs used to generate the segments.

When combined with the project TM, the number of pretranslated segments increases (cf. TM3, TM5, TM6 and TM8-TM11). The best overall results are obtained when using the project TM either in combination with both the new TM generated from the project TM and the new TM gen-

	TM1	TM2	TM3	TM4	TM5	TM6	TM7	TM8	TM9	TM10	TM11
Not started	234	204	150	202	143	143	38	27	**26**	26	**26**
Pre-trans.	155	111	178	108	179	183	139	199	**205**	201	**205**
Fragments	36	110	97	115	103	99	248	199	**194**	198	**194**

Table 5: Overview on the number of segments pre-translated, translated using fragments, or not started.

	TM1	TM2	TM3	TM4	TM5	TM6	TM7	TM8	TM9	TM10	TM11
100%	0	5	5	5	5	5	5	5	**5**	5	**5**
95%-99%	2	3	3	4	4	4	9	9	**9**	9	**9**
85%-94%	0	1	1	1	1	1	2	2	**2**	2	**2**
75%-84%	14	9	14	9	15	15	12	16	**16**	16	**16**
50%-74%	187	130	198	136	202	202	180	223	**226**	226	**226**
No match	142	197	124	190	118	118	137	90	**87**	87	**87**

Table 6: Overview on the number of segments retrieved using the different TMs classified by fuzzy band.

erated from the client's TM (TM9) or combining the three new TMs (TM11). When using the new TM generated from the client TM together with the new product TM (TM10), the number of pretranslated segments decreases slightly (205 → 201), while the number of segments translated using fragments increases slightly (194 → 198).

If we now look at the results obtained in terms of fuzzy matches (cf. Table 6), the same tendencies can be observed. From the very beginning, the number of 100% matches increases from 0 to 5 when using the new TM generated from the project TM. Similarly, the number of matches for the 95–99% fuzzy band also increases (from 2 segments for TM1 up to 9 segments for TM7–TM11), and the 85–94% fuzzy band retrieves a new segment when using the new TM generated from the client TM. The greatest increase in fuzzy matches is experienced by the 50–74% fuzzy band (from 142 segments in TM1 up to 226 in TM9–TM11). Although this band is usually discarded in translation projects as no productivity gains are achieved, an analysis of the new segments retrieved is needed. This would give us potential hints as to what to improve in our new segment generator so that the fuzzy scores are higher. At the same time, it could be the case that our segments are reusable, although the rest of the sentence is not. If this was proven true, a productivity increase may be observed in this band.

In general, it seems that the generation of new TMs using fragments of previously existing segments has a positive effect in the TM fuzzy match retrieval as well as in the generation of translations from fragments that memoQ offers. These positive results indicate that working further on this approach may improve the fuzzy matches and thus enhance the productivity of our translators.

6 Conclusion and future work

In this paper we have explored a new way of generating segments out of previously existing segments. Although we use a naive approach and only make use of punctuation marks and tags to generate such segments, positive results have been obtained in our pilot test. Moreover, as we do not use any type of linguistic information, our approach could be considered language independent.

We are currently working on improving the script that extracts the fragments of segments and generates new ones. This is being done by also analysing in more detail the segments currently retrieved and the segments that could additionally be retrieved. The next step will be to generate yet newer segments by combining the fragments retrieved together and pre-translating files implementing our own "fragment assembly approach" prior to translation. Once this has been done, we will test the final result of our TM population and pre-processing in real projects to measure whether by using this approach translators do translate faster than translating from scratch. This final test will additionally serve as a quality evaluation of the segments newly produced, as we will be able to compare them with the final output produced by translators.

We have also envisioned the combination of already existing methods for retrieving a higher

number of TM matches with our system. Among other approaches, it will be interesting to test the inclusion of paraphrasis and semantic similarity methods to create new TM segments (Gupta and Orăsan, 2014; Gupta et al., 2015).

Finally, another potential application of our approach would be the extraction of terminology databases. In many cases, the segments we extract correspond to terms in the source and target text. A closer analysis of them may give us hints about their properties so that we can filter candidate terms and create terminology databases that can be used in combination with the TMs to translate new projects.

Acknowledgments

The research reported in this paper is supported by the People Programme (Marie Curie Actions) of the European Union's Framework Programme (FP7/2007-2013) under REA grant agreement no. 317471.

References

Kevin Flanagan. 2015. Subsegment recall in Translation Memory – perceptions, expectations and reality. *The Journal of Specialised Translation*, (23):64–88, January.

Rohit Gupta and Orăsan. 2014. Incorporating Paraphrasing in Translation Memory Matching and Retrieval. In *Proceedings of the European Association of Machine Translation (EAMT-2014)*.

Rohit Gupta, Constantin Orăsan, Marcos Zampieri, Mihaela Vela, and Josef van Genabith. 2015. Can Translation Memories afford not to use paraphrasing? In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, Antalya (Turkey), May. EAMT.

Gábor Hodász and Gábor Pohl. 2005. MetaMorpho TM: a linguistically enriched translation memory. In *Proceedings of the workshop: Modern Approaches in Translation Technologies 2005*, pages 25–30, Borovets, Bulgaria.

Ruslan Mitkov and Gloria Corpas. 2008. Improving Third Generation Translation Memory systems through identification of rehetorical predicates. In *Proceedings of LangTech 2008*.

Viktor Pekar and Ruslan Mitkov. 2007. New Generation Translation Memory: Content-Sensitive Matching. In *Proceedings of the 40th Anniversary Congress of the Swiss Association of Translators, Terminologists and Interpreters*.

Michel Simard and Atsushi Fujita. 2012. A Poor Man's Translation Memory Using Machine Translation Evaluation Metrics. In *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas*.

Tom Vanallemeersch and Vincent Vandeghinste. 2015. Assessing linguistically aware fuzzy matching in translation memories. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, Antalya (Turkey), May. EAMT.