Building a Translation Memory to Improve Machine Translation Coverage and Quality

Duncan Gillespie dgillespie@etsy.com
Benjamin Russell brussell@etsy.com
Etsy, Inc.

Abstract

In this paper we discuss the motivation and process for planning, building, and monitoring a translation memory (TM) that serves both human- and machine-translated text segments in a production e-commerce environment. We consider the quality improvements associated with serving human translations for commonly used and mis-translated strings, and the cost benefits of avoiding multiple re-translations of the same source text segments. We cover the technical considerations and architecture for each stage of the TM pipeline, and review the results of using and monitoring the TM in a production setting.

1 Machine Translation at Etsy

Etsy is an online marketplace for handmade and vintage items, specializing in unique goods. It is important our global member base can communicate with one another, even when they speak different languages. Machine translation is a valuable tool for facilitating multilingual interactions on our site and in our apps. An example of the translation interface shown to users can be seen in Figure 1.

Listing descriptions account for the bulk of text we machine translate. We have about 40 million active listings at an average length of around 1,000 characters, with hundreds of thousands of listings created or edited each day. These listings are machine translated into six languages. We also provide machine translation for listing reviews, forum posts, and conversations (messaging between members). For these translations, we send text to a third party machine translation service,[1] with whom we have a fixed monthly budget, imposing a limit on the number of characters we translate per month.

While a user can request a listing translation if we don't already have one (we call this on-demand translation), translating a listing beforehand and showing a visitor the translation automatically (we call this pre-translation) provides a more fluid browsing experience. Pre-translation also allows listings to surface in search results in multiple languages, both for searches on Etsy and on external search engines like Google.

2 The Benefits of a Translation Memory

Many of the strings we machine translate from one language to another are text segments we've seen before. Our most common segments are used in millions of listings, with a relatively small subset of distinct segments accounting for a very large proportion of the content. For example, the sentence "Thanks for looking!" appears in around 500,000 active listings on Etsy, and

[1] Microsoft Translator

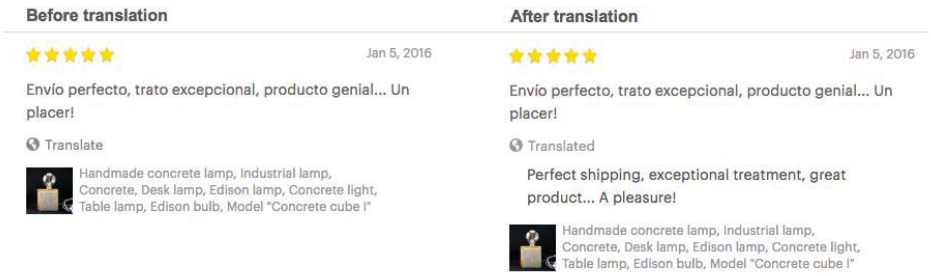

Figure 1: An example review translation on Etsy's website.

has appeared in over 3 million now inactive listings. More broadly, the distribution of unique paragraphs in Etsy listings has a classical Zipfian shape (see Figure 2), with the top segments appearing on the order of 10^7 times and approximately 10^9 distinct segments.

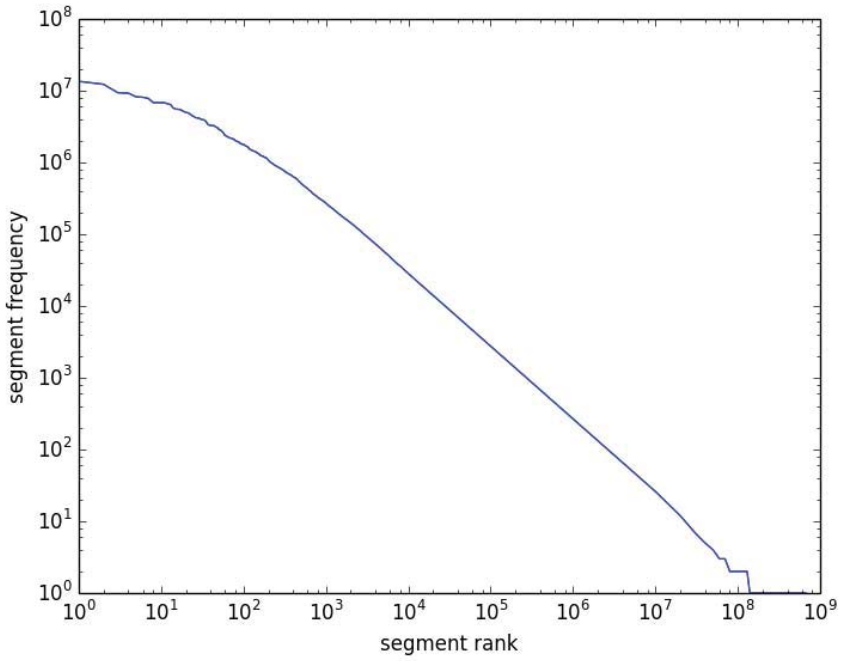

Figure 2: Frequency and rank of text segments (titles, tags, and description paragraphs) appearing in listings on Etsy. The distribution of segments roughly conforms to a Zipfian shape, where a string's rank is inversely proportional to its frequency.

Prior to undertaking this project, a single text segment that appeared in thousands of listings on Etsy was re-translated once for every listing. It would also be re-translated any time a seller edited a listing. This meant our translation budget was being spent on millions of repeat translations that would be better used to translate unique content into more languages.

To solve this problem, we built a translation memory. At its simplest, a translation memory stores a text segment in one language and a corresponding translation of that segment in another language. Translation memories are traditionally used by human translators to avoid re-translating the same text segments multiple times, and to ensure consistency of translations, thereby improving quality (Christensen and Schjoldager, 2010; Reinke, 2013). Combining machine translation with human-translated strings in a translation memory has also been a topic of research, and has been shown to have a positive effect on overall translation quality (Marcu, 2001; Koehn and Senellart, 2010).

For our purposes, storing strings in a translation memory allows us to serve translations for these strings from our own databases, rather than making repeated requests to the translation service. Storing these translations for later reuse has important implications on both quality and coverage. In terms of quality, the translation memory allows us to translate individual strings instead of translating one block of text as a whole. This means we can see which text segments are most commonly used on Etsy and have these segments human translated. Serving human translations instead of machine translations for these common segments improves the overall quality of our translations.

Secondly, storing common translations in the translation memory and serving them ourselves also allows us to drastically reduce the number of duplicate segments we send to the translation service. This process lets us translate seven times more content for the same cost, increasing our overall language coverage.

3 Initial Considerations

We had two main concerns when planning the translation memory architecture. First, we had to plan for adequate **capacity**. The more text segments we store in the translation memory, the greater our coverage. However, storing every paragraph from each of our more than 35 million active listings, and a translation of that paragraph for each of our supported languages, would mean an exceptionally large database table. We wanted to keep the table limit under a billion rows to make sure it was maintainable under our existing MySQL infrastructure.

Second, we needed to provide a mechanism for periodic **deletions**. The translation service's quality is continually improving, and to take full advantage of these improvements we need to periodically refresh entries in the translation memory by deleting older translations. We wanted to be able to delete several hundred million rows on a monthly basis without straining system resources.

4 The Translation Memory Architecture

The translation memory consists of several separate services, each handling different tasks. The services act sequentially upon a given text segment, only sending segments to the third party service after exhausting all other possible translation sources. A full diagram of the pipeline is shown in Figure 3. A brief overview of each step:

4.1 Splitting into segments

The first step of the translation pipeline is splitting blocks of text into individual segments. The two main choices here were splitting by sentence or splitting by paragraph. We chose the latter for a few reasons. Splitting by sentence gave us more granularity, but our estimated translation memory hit rate was only 5% higher with sentences versus paragraphs. The increased hit rate wasn't high enough to warrant the extra logic needed to split by sentence, nor the increase in table rows needed to store every sentence, instead of just every paragraph. Moreover, although automatic sentence boundary detection systems can be quite good, Read et al. (2012) evaluated the most popular systems on user-generated content and found that accuracy peaked at around

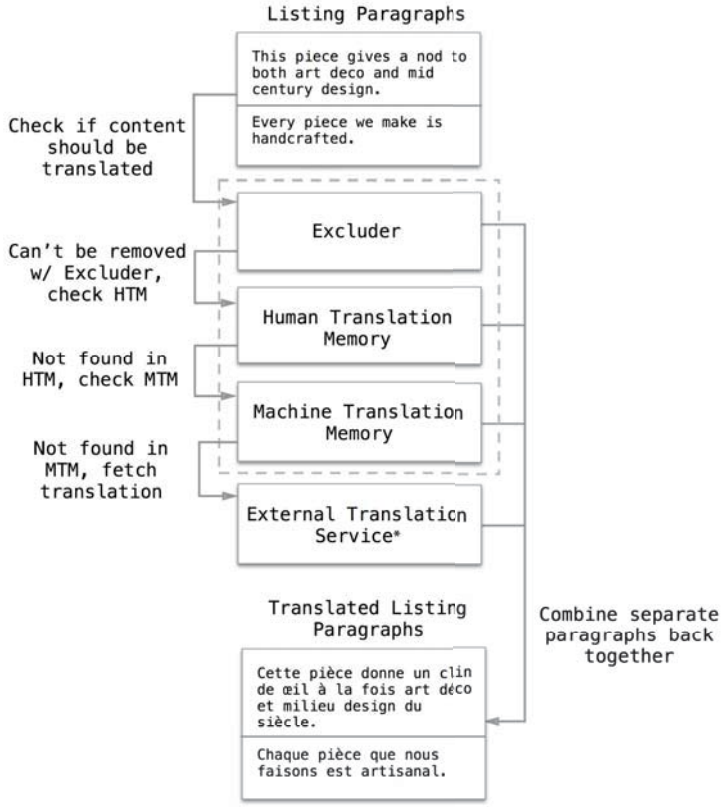

Figure 3: An overview of the translation memory pipeline. The external translation service is Microsoft Translator.

95%. In contrast, using newline characters to split paragraphs is a straightforward and error-free way to segment text.

4.2 Excluder

The Excluder is the first service we use to process translations. It removes any content we don't want to translate, specifically lines containing only links, numbers, or non-alphanumeric characters.

4.3 Human Translation Memory (HTM)

After excluding non-translatable strings, and before looking for a machine translation, we check first for an existing human translation. Human translations are provided by Etsy's professional translators (the same people who translate Etsy's static site content). These strings are stored in a separate table from the Machine Translation Memory and are updated using an internal tool we built, pictured in Figure 4.

4.4 Machine Translation Memory (MTM)

We use sharded MySQL tables to store our machine translation entries. Sharded tables are a well-established pattern at Etsy, and the system works especially well for handling the large row

Figure 4: The interface for managing human-translated segments.

count needed to accommodate the text segments. As mentioned earlier, we periodically want to delete older entries in the MTM to clear out unused translations, and make way for improved translations from the translation service. We partition the MTM table by date to accommodate these bulk deletions. Partitioning allows us to quickly drop all the translations from a certain month without worrying about straining system resources by deleting millions of individual entries.

4.5 External Translation Service

If there is new translatable content that doesn't exist in either our HTM or MTM, we send it to the translation service. Once translated, we store the segment in the MTM so it can be used again later.

4.6 Re-stitching segments

Once each of the segments has been processed by one of our four services, we stitch them all back together in the proper order.

5 The Results

We implemented the Excluder, HTM, and MTM in that order. Implementing the Excluder first allowed us to refine the text splitting, restitching, and monitoring aspects of the pipeline before worrying about data access. Next we built the HTM and populated it with several hundred translations of the most common terms on Etsy. Finally, at the end of November 2015, we began storing and serving translations from the MTM.

5.1 Coverage

As you can see from the graphs in Figure 5, we now only send out 14% of our translations to the translation service, and the rest we can handle internally. Practically, this means we can pre-translate over seven times more text on the same budget. Prior to implementing the translation memory, we pre-translated all non-English listings into English, and a majority of the rest of our listings into French and German. With the translation memory in place, we are pre-translating all eligible listings into English, French, German, Italian, Spanish, and Dutch, with plans to scale to additional languages.

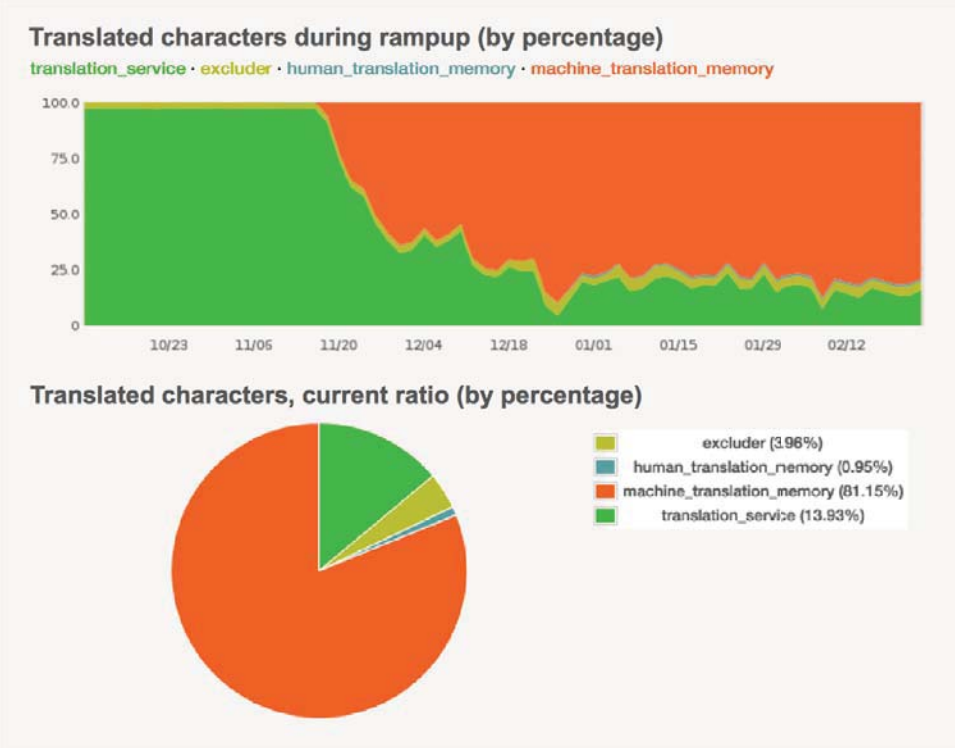

Figure 5: With the translation memory in place, we only need to send out a fraction of the listing segments to the third-party service for re-translation.

5.2 Quality

1% of our translations (by character count), are now served by the human translation memory. These HTM segments are mostly listing tags. These tags are important for search results and are easily mis-translated by an MT system because they lack the context a human translator can infer more easily. Additionally, human translators are better at conveying the colloquial tone often used by sellers in their listing descriptions. With the HTM in place, the most common paragraph on Etsy, “Thanks for looking!” is human translated into the friendlier, “Merci pour la visite !” rather than the awkward, “Merci pour la recherche !” The English equivalent of this difference would be, “Thanks for visiting!” versus “Thanks for researching!”

5.3 Monitoring

Since a majority of our translation requests are now routed to the MTM rather than the third-party translation service, we monitor our translations to make sure they are sufficiently similar to those served by the translation service. To do this, we sample 0.1% of the translations served from the MTM and send an asynchronous call to the translation service to provide a reference translation of the string. Then we log the similarity (the percentage of characters in common) and Levenshtein distance (also known as edit distance) between the two translations. As shown in Figure 6, we track these metrics to ensure the stored MTM translations don’t drift too far from the original third party translations.

For comparison, as you can see in Figure 7, the similarity for HTM translations is not as

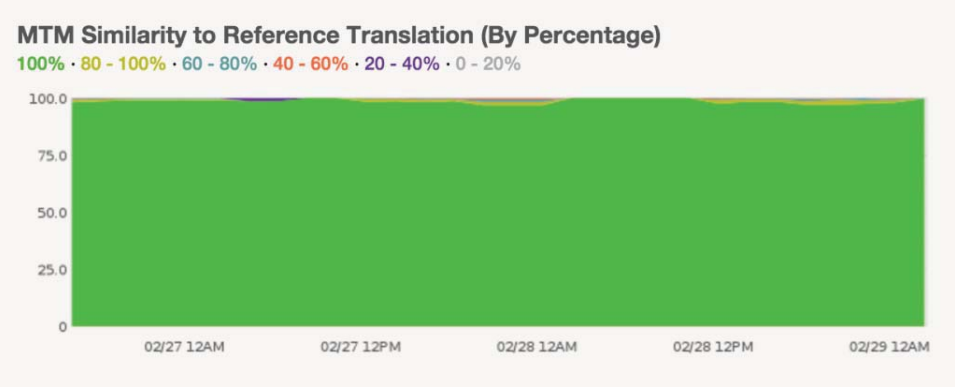

Figure 6: Tracking translation drift allows us to understand the difference between the translations we serve internally and the translations we would get from using the third party translation service. Too much drift means we are not refreshing the translations often enough.

high, reflecting the fact that these translations were not originally drawn from the third party translation service.

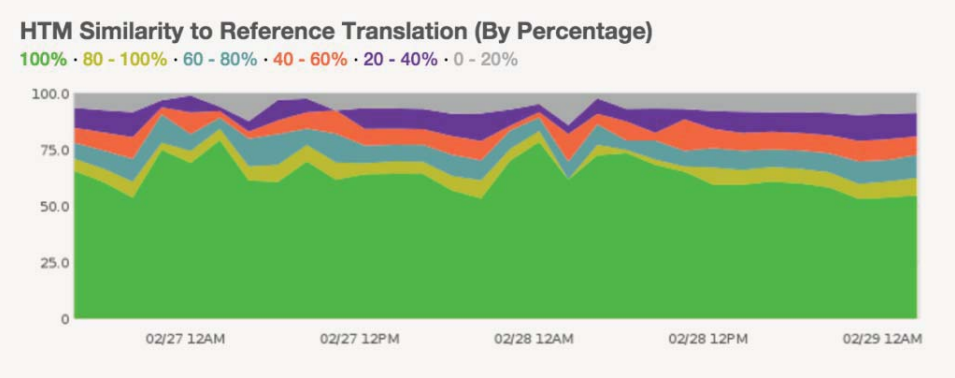

Figure 7: The translations from our human translation are expected to be significantly different than the machine translations we would be serving otherwise.

6 Additional Benefits

6.1 Correcting mis-translations

Statistical machine translation engines are trained on large amounts of data, and sometimes this data contains mistakes. The translation memory gives us more granular control over the translated content we serve, allowing us to override incorrect translations while the translation service we use works on a fix. In Figure 8 is an example where “Realistic bird” is mis-translated into German as “Islamicrevolutionservice.”

With the translation memory, we can easily correct problematic translations like this by adding an entry to the human translation memory with the original listing title and the correct German translation.

Figure 8: Correcting mis-translations quickly is important to maintain the trust of our buyers.

6.2 Respecting sellers paragraph choices

Handling paragraph splitting ourselves has the additional benefit of improving the quality of translation for many of our listings. Etsy sellers frequently include lists of attributes and other information without punctuation in their listings. For example, a listing description might contain the following three lines:

Dimensioni 24 × 18 cm
Spedizione in una scatola protettiva in legno
Verr fornito il codice di monitoraggio (tracking code)

The translation service often combines these lists into a single sentence, producing a translation like this:

Size 24 × 18 cm in a Shipping box wooden protective supplies the tracking code (tracking code)

By splitting on paragraphs, our sellers' choice of where to put line breaks is now always retained in the translated output, generating a more accurate (and visually appealing) translation like this:

Size 24 × 18 cm
Shipping in a protective wooden box
You will be given the tracking code (tracking code)

Splitting on paragraphs prior to sending strings out for translation is an improvement we could have made independent of the translation memory, but it came automatically with the infrastructure needed to build the pipeline.

7 Conclusion and Future Work

Greater accuracy for listing translations means buyers can find the items they are looking for more easily, and sellers' listings are more faithfully represented when translated. The translation memory allows us to bring this internationalized Etsy experience to more users in more languages, making it easier to connect buyers and sellers from around the world. It also helps our return on investment in machine translation, since we are able to translate more content with the same budget.

To further the translation memory quality improvements, we recently started using an Etsy-customized engine built on top of the third-party translation service's generic engine. We saw

a significant increase in purchase rate from users who interacted with the customized engine instead of the generic engine (Russell and Gillespie, forthcoming 2016).

Future efforts will be focused on improving translation quality in our search experience. Serving more human translations instead of machine translations will be especially important in search, where queries are often short strings lacking context. Quality improvements in search ensure all users have access to listings in our global marketplace, regardless of mismatches between query language and listing language.

8 Acknowledgements

We'd like to thank product manager Ray Flournoy for his guidance and input during the planning and construction of the translation memory.

References

Christensen, T. P. and Schjoldager, A. (2010). Translation-memory (TM) research: What do we know and how do we know it. *Hermes*, 44:89–101.

Koehn, P. and Senellart, J. (2010). Convergence of translation memory and statistical machine translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, pages 21–31.

Macklovitch, E. and Russell, G. (2000). What's been forgotten in translation memory. In *Conference of the Association for Machine Translation in the Americas*, pages 137–146. Springer.

Marcu, D. (2001). Towards a unified approach to memory- and statistical-based machine translation. In *Proceedings of the 39th annual meeting on association for computational linguistics*, pages 386–393. Association for Computational Linguistics.

Read, J., Dridan, R., Oepen, S., and Solberg, L. J. (2012). Sentence boundary detection: A long solved problem? In *COLING (Posters)*, pages 985–994.

Reinke, U. (2013). State of the art in translation memory technology. *Translation: Computation, Corpora, Cognition*, 3(1):27–48.

Russell, B. and Gillespie, D. (forthcoming 2016). Measuring the behavioral impact of machine translation quality improvements with A/B testing. In *Conference on Empirical Methods in Natural Language Processing*.