Improve MT for Search with selected Translation Memory using Search Signals

Bryan Zhang bryzhang@amazon.com
Amazon.com

Abstract

Multilingual search is indispensable for a seamless e-commerce experience. E-commerce search engines typically support multilingual search by cascading a machine translation step before searching the index in its primary language. In practice, search query translation usually involves a translation memory matching step before machine translation. A translation memory (TM) can reduce the computation footprint in production, enforce certain terminology translation and enable us to fix translation issues quickly. In this study, we propose (1) a method of improving MT query translation using such TM entries when the TM entries are only substrings of a customer search query, and (2) an approach to selecting TM entries using search signals that can contribute to better search results.

1 Introduction

Localization of e-commerce sites has led users to expect search engines to handle multilingual queries and return product information in customers' preferred language. Multilingual product search capability is essential for modern e-commerce product discovery (Lowndes and Vasudevan, 2021). Recent proposals of cross-lingual information retrieval that handle multilingual queries and language-agnostic cross-border product indexing have gained traction with neural search engines (Hui et al., 2017; McDonald et al., 2018; Nigam et al., 2019a; Lu et al., 2021; Li et al., 2021), but legacy e-commerce search indices are still built on monolingual product information and support for multilingual search is bridged using machine translation (*Search MT*) (Nie, 2010; Rücklé et al., 2019; Saleh and Pecina, 2020; Bi et al., 2020; Jiang et al., 2020). In practice, a translation memory matching step is usually arranged before machine translation systems for search query translation.

A translation memory (TM) is a database which stores the source text and its corresponding translation in language pairs that have been previously translated. For example, *rasierwasser*$\rightarrow$ *aftershave*, *kinder schokolade* $\rightarrow$ *kinder chocolate* are entries for German-English translation memory. The translation memory is usually activated when a run-time query exactly matches an entry in the memory. Therefore, a translation memory can (i) reduce the computation footprint and latency for synchronous translation (ii) effectively enforce terminologies for specific brands or products. Although such issues can be mitigated through terminology constraint mechanism in the machine translation model (Dinu et al. (2019); Post and Vilar (2018); Susanto et al. (2020); Wang et al. (2021); Ailem et al. (2021)), the turnover time to fix the translation would be unacceptable to the users and companies that expect an instant fix and, (iii) fix machine translation issues that cannot be resolved easily or quickly without retraining/tuning the machine translation engine in production (Kanavos and Kartsaklis (2010); Caskey and Maskey (2013); Luo et al. (2022); Tan (2022)).

We have also observed that many translation memory (TM) entries can partially match a large percentage of queries at run-time. It is necessary to integrate translation memory (TM) to the machine translation systems and enable a run-time query to partially match an entry in the memory, that way one query translation can come from both translation memory and machine translation systems. Unlike exact string matching, one TM entry can only impact one run-time query, partial matching can allow one TM entry to impact a large number of queries, so it is crucial only to select TM entries that can bring a positive impact to the customers' shopping experience. Therefore, in this paper we propose:

- a method of exploiting the placeholder features of modern industrial machine translation, and implementing a sub-string partial matching feature that enables the NMT models at run-time to recognize the longest TM entry as sub-string, then use the sub-string TM translation to replace the MT output of that sub-string.
- an approach to selecting an optimal translation memory (TM) subset for partial matching using search signals. The selected TM subset can have contribute to better query translation quality and have larger positive impact on the search results

The rest of the paper is organized as following: we will propose the method of integrating translation memory to machine translation systems enabling sub-string matching in section 2; In section 3 we will propose an approach to selecting an optimal translation memory subset using search signal; Section 4 is the experiment setup and section 5 is result and analysis. We draw the conclusion in section 6.

2 Machine translation with selected translation memory in production

This approach includes a sub-string partial matching feature that enables neural machine translation (NMT) models at run-time to recognize the longest TM entry as a sub-string, then use the sub-string TM translation to replace the MT output of that sub-string. Figure 1.illustrates this approach using a query translation example from German to English:

- STEP 1-2: Given a query *rasierwasser tabak*, if there is an entry (or entries) matched to the query as sub-string (s) (e.g. *rasierwasser - aftershave*)[1], the matched sub-string in the source query is replaced with a placeholder. (e.g. *[placeholder_1] tabak*)
- STEP 3: The query with the placeholder will be passed to the machine translation model. The machine translation model returns the query translation with placeholder (e.g. *[placeholder_1] tobacco*)
- STEP 4: The placeholder will be replaced by the translation from the matched entry (e.g. *aftershave tobacco*).

2.1 Sub-string matching

We propose to use a back-off n-gram matching algorithm that will match the translation memory entries in the source language to queries as sub-strings: given a query, the query is first converted into n-grams, then we try to match the n-grams to the entries in the translation memory. We start the value of n as the number of the tokens in the query and then decrease the value of n for the n-grams until $n = 1$ or until we find a match in the memory. This way, we can aim at finding the longest match.

[1] For cases where a term (s) in the source needs to preserved in the translation, the same entry (in the source language) is stored on both source and translation sides

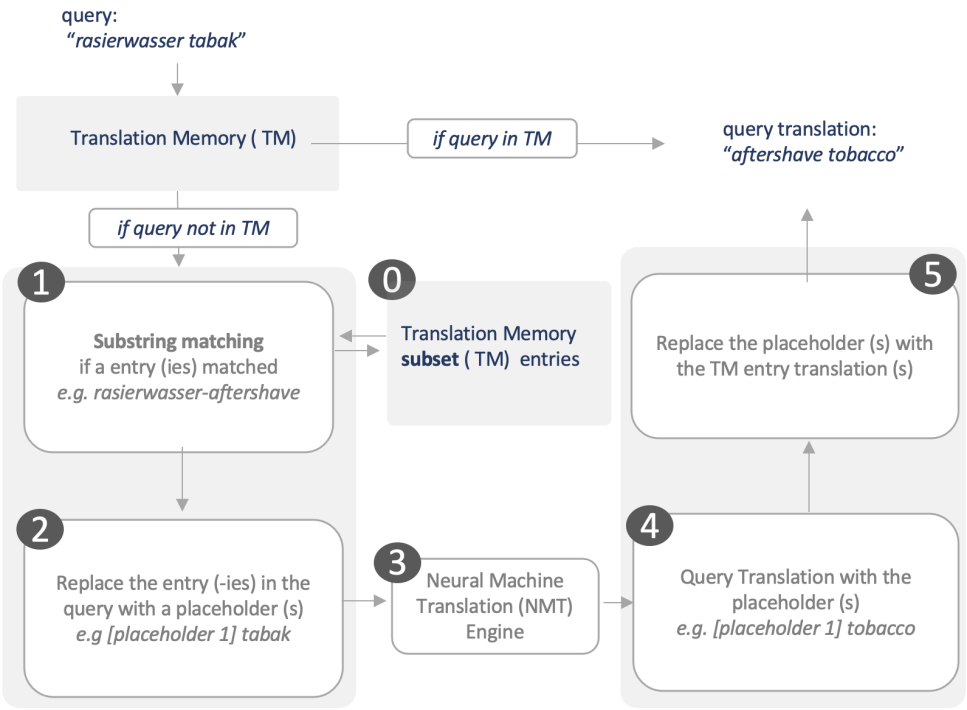

Figure 1: The method of MT with selected TM entry substitution in production

2.2 Augmenting NMT with placeholders

We augment the neural machine translation (NMT) models with placeholder data during the training, so the NMT models can translate queries with placeholders and keep those placeholders intact during the translation process. Those placeholders are also serialized tokens e.g. *placeholder_1, placeholder_2* which are part of the vocabulary used at inference time.

3 Translation memory (TM) subset selection using search signal

Partial matching enables one TM entry to impact a larger number of queries, so it is crucial only to select TM entries that can bring a positive impact to the customers' shopping experience. The search results matter from the customer's perspective and MT query translations are used as intermediate artifacts for search. We rely on customer purchasing behavior as a signal for relevance judgments to automatically estimate the search performance of MT query translations, and a TM entry is selected if the MT query translation with the TM sub-string substitution has better search performance than the default MT query translation.

Figure 2 illustrates our proposed translation memory subset selection workflow. For a given TM entry e, we first sample queries in the source language Q_{src} from the historical traffic data that can partially match to the entry. We then also sample hundreds of thousands of the purchased product IDs P and their frequencies F associated with each source query. We will use the top two most frequent source queries $q_{src}(q_{src} \in Q_{src})$ for selection: for each source query q_{src}, we will use MT to generate two versions of query translation, one t_{mt} is returned from the MT and other t_{mt+tm} is returned from MT with translation memory (TM). We will retrieve two sets of search results R_{mt} and R_{mt+tm} for these two versions of query translations

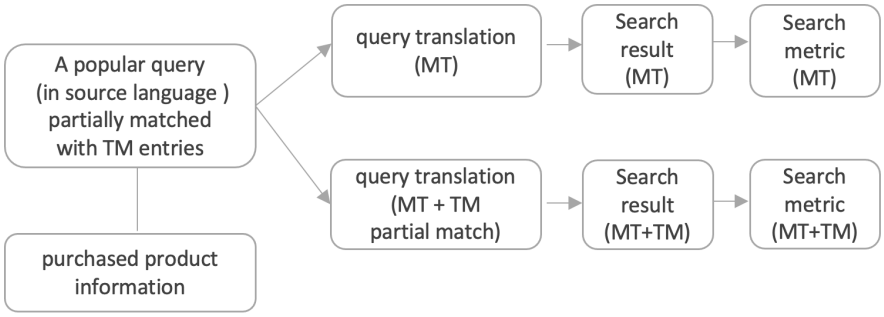

Figure 2: The translation memory entry selection approach

t_{mt} and t_{mt+tm} respectively; then we will use the purchased product IDs P_q associated the source query q_{src} as a proxy to the relevant products and logarithm of their frequencies F_q as the scaled relevance for the search rank-based metrics S computation of these two sets of search results S_{mt} and S_{mt+tm}. The TM entry is selected if the search rank-based metric of the MT query translation with TM S_{mt+tm} is higher than the search rank-based metric of default query MT translation S_{mt} for both source queries.

We also observe some TM entries are terminologies such as brands, and they also overlap with the common vocabulary of the source language that usually needs to be translated. For example, take the entry *kinder*. This word is both a brand and a common word in German meaning *children*; when it refers to the brand it is expected to be preserved in the German-to-English translation. Therefore, if such entries exist in the TM, we suggest creating a frequent collocation *kinder schokolade* alone based on the query log and adding the new entry pair *kinder schokolade - kinder chocolate* to the translation memory before the subset selection.

4 Experiment

We conduct both offline and online experiments for the proposed approaches for Portuguese queries on *Amazon.es*, German queries on *Amazon.com (US)*, and Dutch queries on *Amazon.de*.

Machine Translation models: For each language pair we train a transformer-based ((Vaswani et al., 2017)) MT system that is encoder-heavy (20 encoder and 2 decoder layers) (Domhan et al. (2020)) using the Sockeye MT toolkit. We use a vocabulary of 32K BPE (Sennrich et al. (2016)) tokens. We optimise using ADAM (Kingma and Ba (2015)) and perform early-stopping based on perplexity on a held-out dev set. We train on internal general out-domain news data and fine-tune on human translated search queries and synthetically generated query translations through back-translation.

Selected translation memory: Based on the proposed translation memory subset selection workflow from section 3, we have used search rank-based metric *nDCG* (normalized Discounted Cumulative Gain) on the top 16 product search result (*nDCG@16*) as search signal. We have selected approximate 30 thousands translation memory entries for each one of the following language pairs: *nlnl-dede, ptpt-eses* and *dede-enus*.

Test sets: For each language pair, we have sampled 2500 test cases from the query data which has been previously sampled for the translation memory selection. Each test case includes (1)

a query in the source language and (2) purchased product IDs and (3) respective frequencies, and is not used in the TM subset selection. And the source query in each test case can partially match a unique entry from the selected TM.

Metric hyper-parameters for evaluation: We set K to 16 for the top-k search results, using the top-16 products in the search results to compute *nDCG* (normalized discounted cumulative gain), *MAP* (mean average precision) and *MRR* (mean reciprocal rank) (Järvelin and Kekäläinen, 2002; Wu et al., 2018; Nigam et al., 2019b).

5 Results and analysis

Table 1 presents the offline evaluation metrics *nDCG, MAP* and *MRR*. All the search metrics have been scaled from 0-1 to 0-100 for convenience. Based on the results, query translations from MT using selected TM have much bigger improvement than the original query translation from MT consistently cross the three language pairs. It suggests the matched sub-strings in the query are translated better with the translation from selected translation memory, and brand-like terms in query translation are also handled properly. For example, with the German-English TM entry *haus laboratories - haus laboratories* in the selected TM, the brand in the source query *haus laboratories lippenstift* is preserved in the query translation *haus laboratories lip stick* whereas the original MT query translation is *house laboratories lip stick*. Table 2 shows more examples of improved query translations using our proposed approaches. It shows both of our proposed approaches are effective, and the query machine translation as a component has much bigger positive impact on the search ecosystems.

		MAP@16	MRR	nDCG@16
German-English	MT	44.2	50.3	51.0
	MT + TM	63.7	75.3	67.5
Dutch-German	MT	47.7	54.1	53.7
	MT + TM	68.7	79.8	70.5
Portuguese-Spanish	MT	15.6	18.2	23.5
	MT + TM	37.7	45.3	49.7

Table 1: Search metrics of two versions of query translations for 3 language pairs

Query (German)	**Query Translation (English) Default MT**	**Query Translation (English) MT with selected TM subset**	**Translation Memory (TM)**
happy hippos kinder schokolade	happy hippos kids chocolate	happy hippos kinder chocolate	kinder schokolade → kinder chocolate
uhren herren patek philippe	watches for men patek philip	watches for men luxury patek philippe	patek philippe → patek philippe
haus laboratories lippenstift	house laboratories lip stick	haus laboratories lip stick	haus laboratories → haus laboratories
game of thrones staffel 8	game of thrones relay 8	game of thrones series 8	game of thrones staffel → game of thrones series
rasierwasser tabak	shaving water tobacco	aftershave tobacco	rasierwasser → aftershave
morgenmantel damen japanisch	morning coat women japanese	dressing gown womens japanese	morgenmantel damen → dressing gown womens
leinwände set	linen set	canvases set	leinwände → canvases
würfelbecher leder	cube cup leader	dice cup leather	würfelbecher → dice cup

Table 2: Examples of MT query translation with and without selected translation memory subset

A/B testing: We have also conducted parallel online A/B testing for the three language pairs. For each language pair, we have deployed a baseline MT and an improved MT model with selected translation memory (TM). Both are integrated into the search pipeline for the designated store. The A/B testing lasted for 4 weeks on average for all the experiments. The improved MT with TM of three language pairs impacted 3-5% of query traffic, and have seen large increases in business metrics, such as, Order Product Sales (OPS), composite contribution profit (CCP), compared to the baseline MT models. Moreover, they all have much larger positive impact on the search result quality, which indicates that our approach has the overall user's multilingual search experiences have received larger improvements.

6 Conclusion

In this paper, we have proposed a method of improving MT query translation using such translation memory (TM) entries when the TM entries are only sub-strings of a customer search query, and an approach to selecting TM entries using search signals that can contribute to better search results. We have conducted both offline and online experiments for improving MT with the selected TM subset using the search signal for Portuguese queries on Amazon.es, German queries on Amazon.com (US), and Dutch queries on Amazon.de. Both off-line and on-line results have shown our approach can improve search query translation and have seen increased order product sales and improved user experience in the multilingual e-commerce search.

References

Ailem, M., Liu, J., and Qader, R. (2021). Encouraging neural machine translation to satisfy terminology constraints. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1450–1455, Online. Association for Computational Linguistics.

Bi, T., Yao, L., Yang, B., Zhang, H., Luo, W., and Chen, B. (2020). Constraint translation candidates: A bridge between neural query translation and cross-lingual information retrieval.

Caskey, S. P. and Maskey, S. (2013). Translation cache prediction.

Dinu, G., Mathur, P., Federico, M., and Al-Onaizan, Y. (2019). Training neural machine translation to apply terminology constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3063–3068, Florence, Italy. Association for Computational Linguistics.

Domhan, T., Denkowski, M., Vilar, D., Niu, X., Hieber, F., and Heafield, K. (2020). The sockeye 2 neural machine translation toolkit at AMTA 2020. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 110–115, Virtual. Association for Machine Translation in the Americas.

Hui, K., Yates, A., Berberich, K., and de Melo, G. (2017). PACRR: A position-aware neural IR model for relevance matching. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1049–1058, Copenhagen, Denmark. Association for Computational Linguistics.

Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Jiang, Z., El-Jaroudi, A., Hartmann, W., Karakos, D., and Zhao, L. (2020). Cross-lingual information retrieval with BERT. In *Proceedings of the workshop on Cross-Language Search and Summarization of Text and Speech (CLSSTS2020)*, pages 26–31, Marseille, France. European Language Resources Association.

Kanavos, P. and Kartsaklis, D. (2010). Integrating machine translation with translation memory: A practical approach. In *Proceedings of the Second Joint EM+/CNGL Workshop: Bringing MT to the User: Research on Integrating MT in the Translation Industry*, pages 11–20, Denver, Colorado, USA. Association for Machine Translation in the Americas.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. volume abs/1412.6980.

Li, S., Lv, F., Jin, T., Lin, G., Yang, K., Zeng, X., Wu, X.-M., and Ma, Q. (2021). Embedding-based product retrieval in taobao search. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3181–3189.

Lowndes, M. and Vasudevan, A. (2021). Market guide for digital commerce search.

Lu, H., Hu, Y., Zhao, T., Wu, T., Song, Y., and Yin, B. (2021). Graph-based multilingual product retrieval in E-commerce search. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pages 146–153, Online. Association for Computational Linguistics.

Luo, C., Lakshman, V., Shrivastava, A., Cao, T., Nag, S., Goutam, R., Lu, H., Song, Y., and Yin, B. (2022). Rose: Robust caches for amazon product search. In *The Web Conference 2022*.

McDonald, R., Brokos, G., and Androutsopoulos, I. (2018). Deep relevance ranking using enhanced document-query interactions. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1849–1860, Brussels, Belgium. Association for Computational Linguistics.

Nie, J.-Y. (2010). Cross-language information retrieval. *Synthesis Lectures on Human Language Technologies*, 3(1):1–125.

Nigam, P., Song, Y., Mohan, V., Lakshman, V., Ding, W. A., Shingavi, A., Teo, C. H., Gu, H., and Yin, B. (2019a). Semantic product search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, KDD '19, page 2876–2885, New York, NY, USA. Association for Computing Machinery.

Nigam, P., Song, Y., Mohan, V., Lakshman, V., Weitian, Ding, Shingavi, A., Teo, C. H., Gu, H., and Yin, B. (2019b). Semantic product search.

Post, M. and Vilar, D. (2018). Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324, New Orleans, Louisiana. Association for Computational Linguistics.

Rücklé, A., Swarnkar, K., and Gurevych, I. (2019). Improved cross-lingual question retrieval for community question answering. In *The World Wide Web Conference*, WWW '19, page 3179–3186, New York, NY, USA. Association for Computing Machinery.

Saleh, S. and Pecina, P. (2020). Document translation vs. query translation for cross-lingual information retrieval in the medical domain. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6849–6860, Online. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Susanto, R. H., Chollampatt, S., and Tan, L. (2020). Lexically constrained neural machine translation with Levenshtein transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3536–3543, Online. Association for Computational Linguistics.

Tan, L. (2022). Tmnt:translation memory and neural translation.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, K., Gu, S., Chen, B., Zhao, Y., Luo, W., and Zhang, Y. (2021). TermMind: Alibaba's WMT21 machine translation using terminologies task submission. In *Proceedings of the Sixth Conference on Machine Translation*, pages 851–856, Online. Association for Computational Linguistics.

Wu, L., Hu, D., Hong, L., and Liu, H. (2018). Turning clicks into purchases: Revenue optimization for product search in e-commerce. SIGIR '18, page 365–374, New York, NY, USA. Association for Computing Machinery.