Query Transformation for Multi-Lingual Product Search

Qie Hu
Amazon

Hsiang-Fu Yu
Amazon

Vishnu Narayanan
Amazon

Ivan Davchev
Amazon

Rahul Bhagat
Amazon

Inderjit S. Dhillon
Amazon & UT Austin

ABSTRACT

In this paper, we study the problem of enabling multi-lingual product search for a global shopping store. In particular, given an existing search system and product catalog in a primary language, and a search query in a secondary language, transform the query into a *semantically equivalent* one in the primary language in order to retrieve the most relevant products. Direct application of machine translation does not always work well in this application due to several factors: 1) lack of consideration of the search system's response to a transformed search query, 2) sensitivity to spelling/grammatical errors, 3) fragility to inputs in a language that is different from the ones the search system is trained for, and 4) poor handling of named entities (e.g. brand names, model numbers). To address these challenges, we propose a Query Transformation system that consists of 1) a language identifier to detect the language of the input query, 2) a deep neural machine translation model fine-tuned on human-curated parallel query corpus and learned, during training, to copy entities such as model numbers, and 3) a traffic re-ranker which selects the transformation that may help the search system retrieve the most relevant products. Furthermore, we show that standard machine translation evaluation metrics such as BLEU are unsuitable for this application. Therefore, we propose a new offline performance metric that measures how accurately a transformed query reflects customer's shopping intent and how well the existing search system responds to the transformed query. We present compelling offline and online results: 11% and 3% in improvements in offline nDCG@8 for Spanish (ES) ⟶ English (EN) and French (FR) ⟶ EN, and 10% and 22% in reduction in online product type search defects for ES⟶EN and FR⟶EN, respectively, over a state-of-the-art statistical machine translation system for product search.

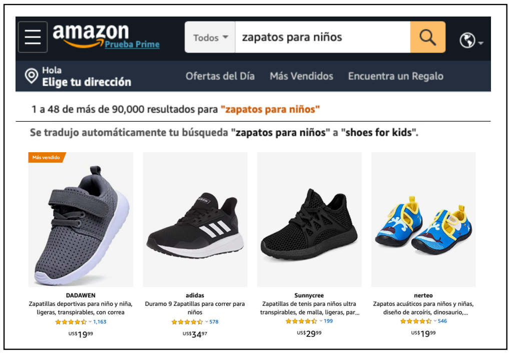
KEYWORDS

Machine Translation, Product Search

ACM Reference Format:
Qie Hu, Hsiang-Fu Yu, Vishnu Narayanan, Ivan Davchev, Rahul Bhagat, and Inderjit S. Dhillon. 2020. Query Transformation for Multi-Lingual Product Search. In *San Diego '20: SIGKDD, August 22–27, 2020, California, CA.* ACM, New York, NY, USA, 9 pages.

1 INTRODUCTION

Multi-lingual product search is an important component for global shopping stores with customers speaking various native languages. With multi-lingual product search, customers can search and shop using their language of preference. Figure 1 shows that customers in the United States (U.S.) can search and shop using Spanish on Amazon.com.



zapatos para niños

1 a 48 de más de 90,000 resultados para "zapatos para niños"

Se tradujo automáticamente tu búsqueda "zapatos para niños" a "shoes for kids".

amazon
Prueba Prime
Todos
Hola
Elige tu dirección
Ofertas del Día
Más Vendidos
Encuentra un Regalo
Más vendido
DADAWEN
Zapatillas deportivas para niño y niña, ligeras, transpirables, con correa
1,163
US$19.99
adidas
Duramo 9 Zapatillas para correr para niños
578
US$34.97
Sunnycree
Zapatillas de tenis para niños ultra transpirables, de malla, ligeras, par...
199
US$29.99
nerteo
Zapatos acuáticos para niños y niñas, diseño de arcoíris, dinosaurio,...
546
US$19.99

Figure 1: Snapshot of searching and shopping on U.S. Amazon.com online store using Spanish query "zapatos para niños".

However, enabling multi-lingual search experience on top of an existing search system is challenging due to the huge scale of the catalog: billions of products exist in the catalog. Due to infrastructure constraints, it is neither feasible nor practical to re-index the entire catalog with translated descriptions and attributes in different languages.

Instead, as shown in Figure 2, we consider a more scalable query transformation approach that fully utilizes the existing search system: given a query in a secondary language, we first transform it to a *semantically equivalent* query in the primary language used in the existing system. This transformed query is then sent to the existing search system to retrieve the results. As product matching and ranking are all done in the primary language, we can retain the existing indexing infrastructure that operates on the primary language, and leverage the high quality search results from the existing state-of-the-art search system. It is worth mentioning that this plug-and-play query transformation approach can be easily extended to offer secondary-language-shopping feature to multiple languages, and would make efficient use of computing infrastructure.

At a first glance, we may think that a standard machine translation model can be applied to transform search queries to enable multi-lingual product search. However, there are many challenges that are specific to query transformation which make this simple

SIGIR eCom'20, July 30, 2020, Virtual Event, China

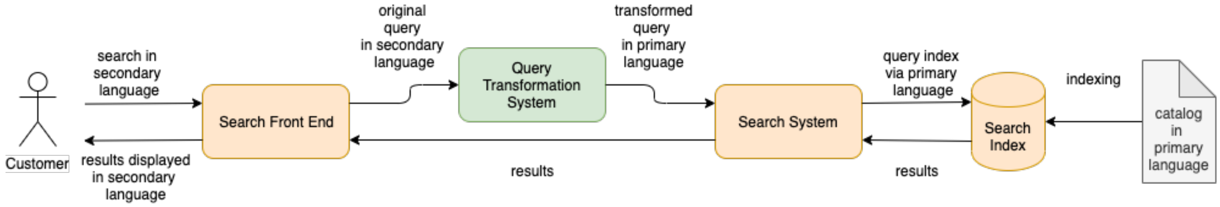

Customer
search in secondary language
results displayed in secondary language
Search Front End
original query in secondary language
Query Transformation System
transformed query in primary language
results
Search System
query index via primary language
results
Search Index
indexing
catalog in primary language

Figure 2: Schematic of query transformation system for multi-lingual search.

approach suboptimal:

- Lack of parallel query corpus. Unlike general machine translation tasks, where there is abundance of manually created parallel corpus from various sources (e.g. Wikipedia, novels, video subtitles), the lack of parallel query corpus makes it harder to train well-performing machine translation models.
- Customer search queries have unique characteristics that distinguish them from general text used in many other natural language processing (NLP) tasks. For instance, queries are usually shorter and formulated with more named entities and spelling/grammatical errors than general text [4]. As a result, it is hard for a pre-trained general purpose machine translator to generate accurate translations for queries.
- Lack of suitable performance metrics. The quality of transformed queries should be measured by how easily customers can find products relevant to their shopping intent. Therefore, the quality depends on both how accurately the transformed query reflects the shopping intent of the original query, as well as how well the existing search system responds to the transformed query. However, most existing metrics for machine translation, such as BLEU and METEOR, are not designed to reflect this [3, 35].

In order to evaluate the quality of the transformed queries to facilitate fast exploration of various modeling innovations, we propose a new family of offline metrics that are driven by customer engagement and are specific to the existing search system. We refer to these metrics as behavior metrics in this paper. Guided by the proposed metrics, we carefully develop and validate various state-of-the-art NLP models and techniques to address the aforementioned challenges for real world product search queries. We are able to build a query transformation system with complete components as illustrated in Figure 3. We further deploy our proposed query transformation system to run online A/B tests on the Amazon store websites in two countries: U.S. and Canada. Both offline and online metrics demonstrate that our proposed query transformation system improves the multi-lingual search experience for Amazon's customers. In this paper, we will share our learnings in the process to build the proposed query transformation system.

This paper is organized as follows. In Section 2, we discuss existing work in neural machine translation (NMT), domain adaptation for low-resource applications and uses machine translation for cross-lingual information retrieval. Section 3 presents our proposed behavior metric and Section 4 describes our proposed Query Transformation system in detail. Offline experiments and online deployment are described in Sections 5 and 6. This is followed by Section 7, which discusses results of ablation studies and shares our learnings from extensive experiments. Finally, we conclude this paper and present some directions for future work in Section 8.

2 RELATED WORK

Recently, NMT systems have demonstrated superior performance, surpassing traditional phrase-based translation models [5], [27], and have become the state-of-the-art in machine translation. NMT systems are sequence to sequence models that typically have an encoder-decoder structure. Different neural architectures have been proposed, including convolutional networks [17, 21, 22] and recurrent networks, such as Long Short-Term Memory, Gated Recurrent Units and Recurrent Neural Networks [7, 18, 32, 39]. The latest sequence to sequence architectures rely on attention mechanisms [1, 36]. Moreover, attention mechanisms have been refined with self-attention [40] and variational attention [2]. In this work, we use a Transformer architecture [40], which relies solely on multi-head self-attention and has achieved state-of-the-art results in many machine translation tasks.

It is well known that to train a high quality NMT model requires large amounts of parallel texts in the source and target languages [43]. However, domain-specific high-quality translations are often scarce and NMT performs poorly in such scenarios. Thus, domain adaptation methods that leverage both out-of-domain and in-domain parallel and monolingual datasets perform an important role in achieving good, domain-specific machine translations. For example, better encoders can be learned using source monolingual data through multitask learning [41]. Target monolingual data, on the other hand, can be used to strengthen the decoder, by using it to train a target language model and fuse it with an NMT model [13, 44]. Another popular approach is back translation, which back translates target sentences into the source language to create a synthetic parallel corpus and incorporate it into training data [10, 14, 38]. Most recently, some researchers have even studied unsupervised machine translation using monolingual corpora only [9, 30], although performances are still not on par with their supervised counterparts. In this paper, we use fine-tuning [11, 19], a proven domain adaptation technique that first pre-trains an NMT model on out-of-domain parallel corpora, followed by fine-tuning the model parameters using in-domain parallel corpora [8, 37].

Most existing machine translation work is focused on translating general texts [6, 23], and there is little published literature on machine translation of search queries. Some earlier works [16, 33, 42] use statistical machine translation (SMT) for cross-lingual information retrieval (CLIR), and more recently, Lignos et al. discuss challenges in optimizing NMT for low-resource CLIR [31]. Our work focuses on query transformation for product search and differs from these studies in two main aspects. First, these studies are based on simulated information retrieval systems on public datasets. We evaluate performance and deploy our system on Amazon.com. Second, we train and test our models using real search queries typed by customers, which tend to be noisier than queries from curated datasets. This makes our problem more challenging.

3 PROPOSED BEHAVIOR METRIC

Machine translation for product search is a task where the end goal is to use the machine translation output for a specific task rather than to have a human read and comprehend. In this case, we wish to find a translation that returns the most relevant products for a search query.

Common metrics used to evaluate machine translation systems, such as BLEU and METEOR [3, 35], are generally based on measuring the degree of overlap of n-grams between a machine translation and a human-generated reference translation. These metrics measure qualities such as accuracy, grammatical correctness and fluency of the translation, which are important in human judgements. As a result, they penalize nuances such as mixing up of singular and plural nouns (e.g. "man" v.s. "men"), incorrect word order (e.g. "pink iPhone" v.s. "iPhone pink") and translation differences (e.g. due to synonyms such as "sneakers" v.s. "trainers"). However, these nuances may not be so relevant in product search.

Therefore, we propose new behavior metrics to directly evaluate machine translation systems based on their performance on the task of multi-lingual product search. The proposed behavior metrics measure how easily customers can find products that are relevant to their shopping intent using a transformed query. More specifically, they quantify the goodness of a transformed query by comparing the set of products retrieved using the transformed query against the set of products purchased by customers who searched using the original query. This assumes that the purchased products associated with the original search query are the oracle, and a better query transformation system is one that produces transformed queries that return more relevant products that customers have indeed purchased. In Section 5.1, we describe how we collect the oracle in this work.

For each test query, we obtain a list of past purchases made using this query and a ranked list of search results returned by a query transformation system's transformed query. Any standard information retrieval evaluation metric can be used to measure the relevance of the search results. In this paper, we use the standard normalized Discounted Cumulative Gain (nDCG) [20]. Given a test query q and its transformation $\tilde{q}$, let $\mathcal{P}(q) := \{p_1, p_2, \ldots, p_n\}$ denote the list of purchased products for q, and let $\mathcal{R}(\tilde{q}) := \{r_1, r_2, \ldots, r_m\}$ denote the ranked list of search results using $\tilde{q}$. We use the following definition of nDCG@$k(q, \tilde{q})$:

$$\begin{aligned} \text{DCG@}k(q, \tilde{q}) &:= \sum_{i=1}^{k} \frac{rel_i}{\log_2(i+1)}, \\ \text{nDCG@}k(q, \tilde{q}) &:= \frac{\text{DCG@}k(q, \tilde{q})}{\text{IDCG@}k(q, \tilde{q})}, \end{aligned} \tag{1}$$

where $rel_i = 1$ if there exists $j \in \{1, 2, \ldots, n\}$ such that $r_i = p_j \in \mathcal{P}(q)$, otherwise $rel_i = 0$. IDCG@k denotes the ideal discounted cumulative gain produced by a perfect ranking algorithm. We report nDCG@x averaged over all queries in the test set.

4 QUERY TRANSFORMATION

In this section, we introduce our proposed query transformation system, QT, for multi-lingual product search.

4.1 Problem Setup and Notations

Let X and Y denote two languages. We assume a bilingual shopping store that uses language Y as the primary language to interface with customers and search the catalog. In addition, customers have the option to explicitly select a secondary language X and shop using language X. For this scenario, we would like to design a query transformation system that transforms search queries from X to the primary language Y in order to search the catalog and return the most relevant search results to the customer.

Let $\mathcal{X}$ and $\mathcal{Y}$ denote the collections of all sentences in languages X and Y, and let Q_X and Q_Y denote the collections of all search queries in the corresponding languages. We assume that there is a large set of out-of-domain parallel sentences $\mathcal{T}_{\text{out}} = \{(x, y)\}$, where $x \in \mathcal{X} \setminus Q_X$, $y \in \mathcal{Y} \setminus Q_Y$. In other words, the sentences are from a domain other than product search queries, e.g. public parallel corpora such as Europarl [26] and ParaCrawl [15]. In addition, we assume that there is a small set of in-domain parallel search queries $\mathcal{T}_{\text{q}} = \{(x, y)\}$, where $x \in Q_X$, $y \in Q_Y$, $|\mathcal{T}_{\text{out}}| \gg |\mathcal{T}_{\text{q}}|$ and $|\mathcal{T}|$ denotes the size of set $\mathcal{T}$. To further guide the query transformation, we also make use of two additional datasets. $\mathcal{D}_X = \{x\}$ and $\mathcal{D}_Y = \{(y, n(y))\}$ are two very large datasets of search queries seen on the shopping store in languages X and Y, respectively. Here, $n(y)$ denotes query y's traffic, which reflects the number of times, aggregated over a long time period, the customers used query y to shop in the online store, . In other words, $x \in Q_X$, $y \in Q_Y$ and $|\mathcal{D}_X| \gg |\mathcal{T}_{\text{q}}|$, $|\mathcal{D}_Y| \gg |\mathcal{T}_{\text{q}}|$.

4.2 System Architecture

Figure 3 shows a schematic of our proposed system's architecture. First, the language identifier identifies the language of the input query: if it is in the primary language Y, then it is kept unchanged and used directly to search for products; on the other hand, if the input query is in the secondary language X, then it is first machine translated into language Y. In the latter case, the machine translation model produces top K candidate translations, which are subsequently re-ranked using query traffic, in order to find the best query transformation for product search. Next, we describe each component in the proposed QT system in detail.

4.2.1 Language Identification. Data analysis reveals that customers use a mixture of search queries in languages X and Y even when

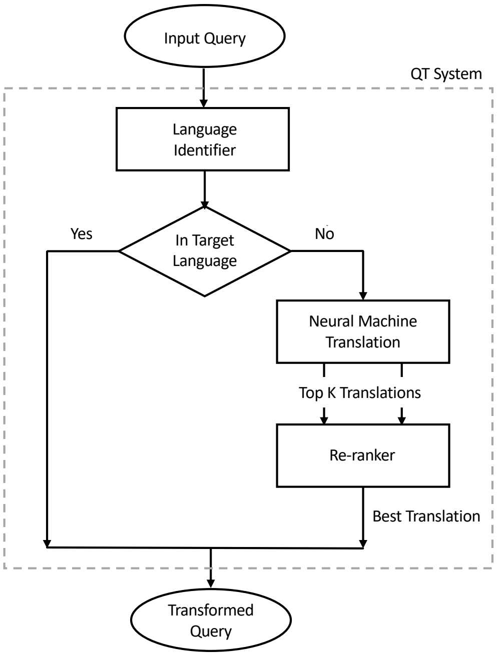


Figure 3: Proposed query transformation (QT) architecture

they choose language X as their language of preference. Therefore we develop a query language identifier to estimate the language of an input query. Direct application of general language identification models on search queries is unable to achieve the performance we require for this application. Again, due to search queries' unique characteristics: short in length and often containing named entities and spelling mistakes, a single query may contain words in multiple languages, etc. Consequently, we built our custom search query language identifier: a Naive-Bayes model using word frequencies calculated from queries in $\mathcal{D}_X$ and $\mathcal{D}_Y$.

4.2.2 Neural Machine Translation. We train an NMT model $f : \mathcal{X} \mapsto \mathcal{Y}$ by minimizing the negative log-likelihood:

$$\mathcal{L}(\theta) = \sum_{(x,y)\in\mathcal{T}} -\log P(y|x;\theta), \tag{2}$$

where θ represents the model parameters and $\mathcal{T}$ represents the training data. We use the Transformer architecture, as they offer state-of-the-art performance in several machine translation tasks.

To alleviate the problem of lack of query specific in-domain training data, we use the fine-tuning approach. The model is first pre-trained on the large out-of-domain dataset, i.e. $\mathcal{T} = \mathcal{T}_{\text{out}}$. We then follow the mixed fine-tuning approach [8]: continue tuning the pre-trained model parameters on a mix of in-domain and out-of-domain corpora. This approach prevents overfitting the model on the small in-domain dataset and mitigates the catastrophic forgetting phenomenon [25]. Here, we choose to fine-tune on a mix of the in-domain parallel query corpus $\mathcal{T}_{\text{q}}$ and an equal number of parallel sentences randomly sampled from the out-of-domain corpus $\mathcal{T}_{\text{out}}$.

In contrast to general text translation, search queries often consist of tokens that contain digits, which may indicate a model number, size or the year that a product is released (example queries in Tables 1 and 2). For these queries, failure to keep these tokens unchanged during machine translation can significantly deteriorate their search results. We propose an approach where the machine translation model learns, at training time, how to copy tokens that contain digits during translation when these tokens appear in the source query. Our approach integrate digit-copy information as inline preprocessing to the query text, and has the advantage of not modifying the original sequence to sequence NMT architecture. As illustrated by examples in Tables 1 and 2, at training time, we replace tokens that contain digits and appear in both source and target queries with special "copy" symbols; and at inference time, we replace all tokens containing digits at the source side prior to passing through the NMT model. We then post-process the output sequence from the NMT model to replace the "copy" symbols with the original tokens.

4.2.3 Traffic Re-ranking. We observe that translation differences that humans may not mind could affect product search performance (e.g. "Nike shoes" v.s. "shoes from Nike") on Amazon.com. In this regard, we use traffic re-ranking to choose a query transformation that may give the best result when used for product search. In particular, the top K candidate translations from the NMT, in the target language Y, are then filtered and re-ranked using $\mathcal{D}_Y$, the search query traffic dataset in the target language. Given an input sequence x, let $z_{\text{mt},i}$ denote the per token likelihood of the i-th translation y_i from the NMT model. Let $\mathcal{H}_K = \{(y_1, z_{\text{mt},1}), (y_2, z_{\text{mt},2}), \ldots, (y_K, z_{\text{mt},K})\}$ denote an ordered list of the top K translations in decreasing order of $z_{\text{mt},i}$. We calcualte the combined score z_i for each y_i:

$$z_i = \begin{cases} z_{\text{mt},i} + \alpha \frac{n_i}{\sum_i n_i}, & \text{if } n_i > 0 \\ -\infty, & \text{if } n_i = 0 \end{cases} \tag{3}$$

where n_i is query y_i's traffic, and α is a tuning parameter that adjusts the relative weights of the likelihood term and the relative traffic term. If all translations have zero search traffic, i.e., $n_i = 0$ for $i = 1, \ldots, K$, then we don't re-rank the translations in $\mathcal{H}_K$ and choose y_1 as the best translation. Otherwise, the translation with the highest score z_i is the final translation from our QT system. As shown in Section 7.1, this approach proves to be effective in our use case. Due to differences in the underlying search engines' matching and ranking algorithms, different re-ranking methods may work better for different shopping platforms.

5 OFFLINE EXPERIMENTS

5.1 Datasets

We carry out offline experiments on two different query translation tasks: ES→EN and FR→EN, where ES, FR and EN are short for Spanish, French and English, respectively. As shown in Table 3, we create an in-domain parallel corpus of approximately 0.2 million pairs of human curated popular search queries for each language pair. We collect tens of millions of out-of-domain parallel sentences for the ES→EN translation task and several millions of out-of-domain parallel sentences for the FR→EN task. Out-of-domain corpora for both language pairs consist of web crawled datasets as

Table 1: Text modification to learn digit-copy during training.

	ES source quey	EN target query
Original query	zapatos para niños talla 6.5	kids shoes size 6.5
Preprocessed	zapatos para niños talla <copy0>	kids shoes size <copy0>

Table 2: Text modification for digit-copy during prediction.

	FR source query	EN target query
Original query	batterie asus x751ld	-
Preprocessed	batterie asus <copy0>	-
NMT output	-	asus <copy0> battery
Post-processed	batterie asus x751ld	asus x751ld battery

well as machine translated parallel texts of product information. We randomly sample 2k pairs of search queries from each in-domain corpora and use as holdout test sets. We use the remaining in-domain and all of out-of-domain corpora for model training. To evaluate the models using our proposed metrics, we record real search queries: 10k queries from Amazon.com in the U.S. when customers use Spanish as their language of preference, and 10k queries from Amazon.ca in Canada when customers use French as their language of preference.

5.2 Preprocessing

Considering the training datasets are noisy, we apply a series of rules to normalize the texts and filter out low-quality sentences. We remove invalid characters, sentences with too many or too few words and pairs of source and target sentences that differ too much in their sentence lengths. In addition to these filtering steps, which are common in machine translation tasks, we also apply standard text normalization steps in information retrieval applications, such as stripping accents, lower-casing and normalizing whitespaces. Unlike general texts, product search queries often contain words that specify quantities such as “2 inches” and “20 pounds”. Often, the units can be expressed in equivalent but alternative ways (e.g. “2 inches”, “2 in” and “2" ”). Hence, we chose to standardize the units (e.g. “2 inches” → “2 in”) and found that this improved the translation performance, perhaps, because in this way, we can use our vocabulary and training data more efficiently to aid the NMT model in translating these units. To compute BLEU scores, we also normalize texts and standardize units at the target side. Calculating the behavior metrics requires no reference translations in the target language.

Our in-domain dataset contains samples whose source and target queries are identical or near identical, and are both in the target language. It has been found that removing such samples from the training data improves the performance of MT models for general translation tasks [34]. Nevertheless, we chose to keep such samples for the reasons below. It is not uncommon for product search queries to contain words from both source and target language. For instance, the query “soulier talon woman” consists of French words “soulier talon” and an English word “woman”. Furthermore, our proposed language identifier is not perfect and thus, queries deemed by the language identifier to be in source language may be in the target language. We found that not discarding training samples with identical or near identical source and target queries improves the translation in these situations, as the model learns to not translate words in the source query that are already in the target language, such as the word “woman” in “soulier talon woman”.

For data modification with “copy” symbols, we do not modify the out-of-domain datasets and use the following rules to replace qualifying words in the in-domain corpus with special “copy” symbols: the word contains at least 4 characters, at least one digit and is in both the source query and the target query. During prediction, we follow the same procedure to normalize the source queries and replace all words containing at least 4 characters and one digit with “copy” symbols.

Both out-of-domain and in-domain corpora are then tokenized using mosesdecoder [27], followed by SentencePiece [28] using joint source and target unigram [29] sub-word segmentation with a vocabulary of 36K tokens.

5.3 Model Details

We use the base Transformer architecture described in [40], with shared source and target embeddings, and decrease the dimension of all inner-layers of the feedforward network sublayers to 1024.

We train all models using label smoothing $\epsilon_{ls} = 0.1$, a batch size of 4000 tokens, Adam optimization [24] with $\beta_1 = 0.9$, $\beta_2 = 0.98$ and an inverted square learning rate schedule with minimum learning rate of 10^{-9}, learning rate warmup over the first 4,000 steps with warmup initial learning rate of 10^{-7}. All models are pre-trained on the out-of-domain corpus for 25 epochs, followed by fine-tuning on the mixed corpus for 8 epochs. For prediction, the NMT uses a beam size of 5 and outputs top 5 translations in the target language.

5.4 Results

We compare our proposed QT system’s performance with two models:

- A state-of-the-art SMT model for Product Search (PS-SMT). We consider this as the baseline.
- AWS Translate, a state-of-the-art NMT model for general translation (AWS Translate).

Table 4 shows the behavior metric and BLEU scores. We report tokenized BLEU and calculate them using 2k human translated parallel search queries for each language pair. We also report the behavior metric nDCG@8 as percentage change over the baseline. Here, the choice of 8 is made because the relevance of the top 8 products found for any search query is one metric that is continuously monitored at Amazon.com. Although absolute values of nDCG@8 are not included here, the reported BLEU scores of both the PS-SMT and AWS Translate demonstrate their competitiveness.

Table 3: Size of training datasets used for ES→EN and FR→EN NMT models and size of test sets used to calculate BLEU and our proposed behavior metric.

	Train (out-of-domain pairs)	Train (in-domain pairs)	Test (human translated pairs)	Test (queries for behavior metric)
ES→EN	$\sim 10^7$	$\sim 200k$	$2k$	10k
FR→EN	$\sim 10^6$	$\sim 200k$	$2k$	10k

Behavior metrics of ES→EN models are evaluated on 10k search queries logged when customers use Spanish as their language of preference. Similarly, FR→EN models are evaluated on 10k search queries when French is the language of preference.

Our proposed QT model achieves the highest behavior metric scores for both ES→EN and FR→EN: increasing nDCG@8 by 11.31% and 3.45%, respectively. Tables 7, 8 and 10 give example query transformations from our QT system, and from the PS-SMT and the AWS Translate models. These examples further demonstrate our proposed QT system's performance.

In addition, observe that there is a lack of correlation between BLEU and the behavior metrics in Tables 4 and 5. For example, the baseline ES→EN model has the highest BLEU 50.1, but lower behavior metric scores than our proposed QT model. The FR-EN AWS Translate model has 38.4 BLEU, higher than the baseline's 35.3. However, the former model's nDCG@8 is -11.65% compared to the baseline.

6 ONLINE DEPLOYMENT

We select the best performing query transformation model based on our offline evaluations, and run online A/B tests. While deploying the model online, we observe that the model's inference time latencies are excessive. To expedite our experiment, we generate large datasets containing millions of pre-computed query transformations from each model and restrict the online experiments to these queries. We run online tests in two stores, Amazon.com in the U.S. and Amazon.ca in Canada. Both stores' search systems and catalogs use English as the primary language. Our experiments target customers who shop in a secondary language of preference, i.e., Spanish on Amazon.com and French on Amazon.ca. In the control group, search results are generated based on the translated query from the baseline, a state-of-the-art SMT system for product search. In the treatment group, search results are generated using our proposed QT system. The search quality between control and treatment groups are evaluated by external human auditors who are native-speakers of Spanish and French, respectively. For each query in the evaluation set, the human auditors label which search results are product-type defects. Given a query, a search result is defined as a product-type defect for this query if it is from a different product category from the product that the query is intended for. For example, showing a pair of shoes as a search result would be a product-type search defect for the query "women's hats". A model that produces fewer product-type search defects is a better model. Our proposed QT system reduced product-type search defects by 10% and 22% in Amazon.com and Amazon.ca, respectively, compared to the state-of-the-art SMT system for product search.

Future work will involve optimizing our QT system's performance to reduce latencies. We will design a system that caches the transformations for popular queries, and only falls back to an online query transformation system for uncommon queries. The idea is that caching of query transformations reduces the overall latency and CPU utilization, while falling back to the online QT system for cache misses increases the coverage of query transformation to uncommon queries as well. In addition, the caching layer allows manual overrides for any truly bad translations and thus, improving overall translation accuracy.

7 DISCUSSION

7.1 Ablation Studies

In this section, we provide a detailed study of the effect of different components of our proposed QT system. Table 4 shows BLEU scores and behavior metrics for our QT system with different ablations. For comparison, we include translations from the PS-SMT and the AWS Translate model in all examples.

Effect of language identification. Table 6 shows that our query language identifier identifies significant percentages of source queries in the test datasets to be English. Removing the language identifier reduces the improvements nDCG@8 7.79% for ES→EN, the second largest negative impact after removing pre-training of the NMT model. For FR→EN, the ablated model without language identifier achieves the lowest nDCG amongst all ablations, only 0.23% better than the baseline. For example, the source query "j1772 charger" shown in Table 7 is in English and refers to electric vehicle chargers. Our language identifier identifies it as an English query and hence, the QT system does not translate it. However, "charger" is also a valid French word meaning "load". Therefore all the systems that do not have a language identifier translate it to "j1772 load", which is a poor translation. As Figure 4 shows, "j1772 load" only returns a single irrelevant product, whereas "j1772 charger" returns 190 products and all top results are relevant.

Effect of traffic re-ranking. Table 8 illustrates the effect of traffic re-ranking. The source query "oppo reno" refers to a cell phone of a specific make and model. All models without traffic re-ranking transformed it into the English query "oppo reindeer". In the QT system, both "oppo reno" and "oppo reindeer" are amongst the top 5 translations from the NMT model. However, since customers have rarely searched using the invalid query "oppo reindeer", this translation is ranked down, and the more popular query "oppo reno" is selected.

Effect of digit-copy. The model without replacing digit-containing words in the training data with "copy" symbols has the smallest reduction in BLEU and behavior metrics, compared to the best model. Diving deeper, we observe that, most of the time, the QT model

Table 4: Comparison of tokenized BLEU and behavior metrics of different models. Values of nDCG@8 are reported as percentage changes over the baseline.

Model	ES→EN BLEU	ES→EN nDCG@8	FR→EN BLEU	FR→EN nDCG@8
Our proposed QT system	49.6	11.31%	39.0	3.45%
AWS Translate	47.8	0.26%	38.4	-11.65%
PS-SMT (Baseline)	50.1	-	35.3	-

Table 5: Comparison of tokenized BLEU and behavior metrics of our proposed QT system and different ablation studies. Values of nDCG@8 are reported as percentage changes over the baseline.

Model	ES→EN BLEU	ES→EN nDCG@8	FR→EN BLEU	FR→EN nDCG@8
Our proposed QT system	49.6	11.31%	39.0	3.45%
QT w/o language identification	48.0	7.79%	39.0	0.23%
QT w/o traffic re-ranking	47.9	9.96%	37.6	1.18%
QT w/o digit-copy	50.6	11.03%	38.9	2.90%
QT w/o pre-training	42.1	-11.44%	37.4	0.44%
QT w/o fine-tuning	49.2	10.28%	36.4	3.03%

Table 6: Percentage of source queries in test datasets that are identified by our language identifier to be in the target language EN.

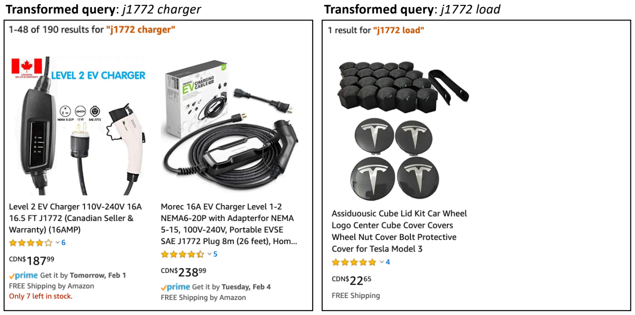
Model	BLEU (2k)	Behavior Metric (10k)
ES→EN	17.9%	32.6%
FR→EN	24.5%	15.5%

Table 7: Example FR→EN translations of the QT system with and without language identifier, and comparison with PS-SMT and AWS Translate.

Source query (FR)	j1772 charger
Reference query (EN)	j1772 charger
QT	j1772 charger
QT w/o language identifier	j1772 load
PS-SMT	j1772 load
AWS Translate	j1772 load

Table 8: Example ES→EN translations of the QT system with and without traffic re-ranking, and comparison with PS-SMT and AWS Translate.

Source query (ES)	oppo reno
Reference query (EN)	oppo reno
QT	oppo reno
QT w/o traffic re-rank	oppo reindeer
PS-SMT	oppo reindeer
AWS Translate	oppo reindeer



Transformed query: *j1772 charger*

1-48 of 190 results for "j1772 charger"

Level 2 EV Charger 110V-240V 16A 16.5 FT J1772 (Canadian Seller & Warranty) (16AMP)
★★★★☆ 6
CDN$18799
prime Get it by Tomorrow, Feb 1
FREE Shipping by Amazon
Only 7 left in stock.

Morec 16A EV Charger Level 1-2 NEMA6-20P with Adapterfor NEMA 5-15, 100V-240V, Portable EVSE SAE J1772 Plug 8m (26 feet), Hom...
★★★★☆ 5
CDN$23899
prime Get it by Tuesday, Feb 4
FREE Shipping by Amazon

Transformed query: *j1772 load*

1 result for "j1772 load"

Assiduousic Cube Lid Kit Car Wheel Logo Center Cube Cover Covers Wheel Nut Cover Bolt Protective Cover for Tesla Model 3
★★★★★ 4
CDN$2265
FREE Shipping

Figure 4: The transformed query with language identification finds more and relevant products.

without digit-copy also keeps digits unchanged during translation. It only fails occasionally, as illustrated in Table 9. Including digit-copy only remedies these occasional failure cases. In particular, observe that the model without digit-copy drops the letter "d" in the model number "x751ld" during transformation, whereas the proposed QT system does not.

Effect of pre-training. Pre-training the NMT model on the out-of-domain data provides the single largest gain in all metrics for the ES-EN QT system: removing it decreases BLEU by 7.5 points and nDCG@8 by 22.75%, compared to the proposed QT system. Similarly, for the FR→EN QT system, removing the NMT model's pre-training stage decreasing nDCG@8 by 3.01% compared to the full QT system. On the other hand, we observe a smaller reduction of 1.6 points in BLEU. Perhaps, due to the out-of-domain corpus used to train the FR→EN NMT model being an order of magnitude smaller than that used for the ES→EN NMT model, qualities of

Table 9: Example FR→EN translations of the QT system with and without digit-copy, and comparison with PS-SMT and AWS Translate.

Source query (FR)	batterie asus x751ld
Reference query (EN)	asus x751ld battery
QT	asus x751ld battery
QT w/o digit-copy	asus x751l battery
PS-SMT	asus x751ld battery
AWS Translate	asus x751ld battery

Table 10: Example FR→EN translations of the QT system with and without fine-tuning, and comparison with PS-SMT and AWS Translate.

Source query (FR)	couche pampers
Reference query (EN)	pampers diaper
QT	pampers diaper
QT w/o fine-tune	pampers layer
PS-SMT	diaper pampers
AWS Translate	pampers layer

the out-of-domain and in-domain datasets, and/or hyperparameter values.

Effect of fine-tuning. On the other hand, removing the fine-tuning stage did not have such a detrimental effect on overall performance. Compared to the baseline, a QT system whose NMT model is not fine-tuned on the query corpus can still achieve 10.28% and 3.03% improvements in nDCG@8 for the ES→EN and FR→EN systems, respectively. Table 10 shows that fine-tuning on the query dataset helps the QT system choose more appropriate translations for polysemic words. In this example, the French word "couche" could mean "layer" and "diaper" depending on the context. Since "pampers" is a brand of baby and toddler products, the two models adapted for product search (our proposed QT system and PS-SMT) correctly translate "couche" to "diaper", whereas the two pre-trained models (QT without fine-tuning and the AWS Translate) translate it less appropriately to "layer".

7.2 Other Attempts and Learnings

We experimented with replacing brand names in the source query with "copy" symbols to prevent brand names from being incorrectly translated by the NMT model. However, brand names can also be common words, such as the fashion brand "bebe", which also means "baby" in Spanish. Therefore, it is uncertain whether the query "t-shirt bebe" means baby t-shirt or a t-shirt from the brand "bebe". Furthermore, spelling errors can make common words appear to be a brand name, e.g. the Spanish queries "perfume dulce" v.s. "perfume dolce". The former means sweet-scented perfumes, while the latter specifically refers to perfumes from the fashion brand Dolce & Gabbana. Mistakenly replacing a common word with the "copy" symbol hence, results in poor query translation. Thus, without the ability to reliably detect brand names in search queries, this approach did not improve the product search performance.

For the NMT component, we also tried the big Transformer architecture in [40], and observed nearly identical results to the base architecture. We hypothesize that this may be due to the short lengths of product search queries, thus making the base Transformer architecture sufficient for our application.

8 CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed our query transformation system for multi-lingual product search for a global shopping store with significant improvements in both offline and online performance metrics. The proposed system is designed to work in synergy with an existing search system and consists of separate modules, each tackling a specific challenge of this problem. We point out that standard machine translation evaluation metrics such as BLEU are unsuitable for this application, and hence, propose a new offline behavior metric that measures how accurately a transformed query reflects customer's shopping intent and how well the existing search system responds to the transformed query. In addition, we discuss our online deployment design and practical trade-offs.

In the future, we hope to incorporate spelling error correction and brand name detection modules, as well as incorporating the use of back-translation [38] to improve the query translation system. We will also continue exploring approaches to speed up the inference of the NMT model.

An interesting area to explore is few-shot adaptation to a new marketplace with a new language. In this scenario, there may not be existing search queries to create a parallel query corpus and no existing customer engagement data to design the traffic re-ranker. What is the best approach to pre-train our query transformation system and then quickly re-train it in order to adapt it for a new language pair?

We also want to experiment with using our query transformation model for other applications. One idea is to use the query transformation model to transform query specific features associated with products. A query transformation model will allow us to generate synthetic queries in a given language from naturally occurring search queries in a different language. These synthetic queries can improve coverage of query specific features. Another

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations.*

[2] Hareesh Bahuleyan, Lili Mou, Olga Vechtomova, and Pascal Poupart. 2017. Variational Attention for Sequence-to-Sequence Models. *CoRR abs/1712.08207* (2017).

[3] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceddings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization.*

[4] Eric Brill and Silviu Cucerzan. 2004. Spelling correction as an iterative process that exploits the collective knowledge of web users. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* 293–300.

[5] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. In *Association for Computational Linguistics.*

[6] Yong Cheng, Lu Jiang, and Wolfgang Macherey. 2019. Robust Neural Machine Translation with Doubly Adversarial Inputs. *CoRR abs/1906.02443* (2019).

[7] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase

Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1724–1734.

[8] Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An Empirical Comparison of Simple Domain Adaptation Methods for Neural Machine Translation. In *Association for Computational Linguistics*. 385–391.

[9] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word Translation without Parallel Data. In *International Conference on Learning Representations*.

[10] Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied Monolingual Data Improves Low-Resource Neural Machine Translation. In *Proceedings of the Second Conference on Machine Translation*. 148–156.

[11] Andrew M. Dai and Quoc V. Le. 2015. Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2018).

[13] Tobias Domhan and Felix Hieber. 2017. Using Target-Side Monolingual Data for Neural Machine Translation Through Multi-task Learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1500–1505.

[14] Sergey Edunov, Mule Ott, Michael Auli, and David Grangier. 2018. Understanding Back-Translation at Scale. In *Association for Computational Linguistics*. 489–500.

[15] Miquel Esplà, Mikel Forcada, Gema Ramírez-Sánchez, and Hieu Hoang. 2019. ParaCrawl: Web-scale parallel corpora for the languages of the EU. In *Proceedings of Machine Translation Summit XVII Volume 2: Translator, Project and User Tracks*. 118–119.

[16] Jianfeng Gao, Jian-Yun Nie, Endong Xun, Jian Zhang, Ming Zhou, and Changning Huang. 2001. Improving Query Translation for Cross-Language Information Retrieval using Statistical Models. In *Proceedings of the* 24th *ACM SIGIR Conference on Research and Development in IR*. 96–104.

[17] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional Sequence to Sequence Learning. In *International Conference of Machine Learning*.

[18] Sepp Hochreiter and Jürgen Schnidhuber. 1997. Long short-term memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[19] Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. In *Association for Computational Linguistics*.

[20] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. 20, 4 (2002).

[21] Lukasz Kaiser, Aidan N. Gomez, and François Chollet. 2017. Depthwise Separable Convolutions for Neural Machine Translation. *CoRR abs/1706.03059* (2017).

[22] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural Machine Translation in Linear Time. *CoRR abs/1610.10099* (2016).

[23] Vladimir Karpukhin, Omer Levy, Jacob Eisenstein, and Marjan Ghazvininejad. 2019. Training on Synthetic Noise Improves Robustness to Natural Noise in Machine Translation. *CoRR abs/1902.01509* (2019).

[24] Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*.

[25] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, Johm Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumarah, and Raia Hadsell. 2017. Overcoming catastrophic forgetting in neural networks. In *Proceedings of the National Academy of Sciences*. 201611835.

[26] Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *MT Summit*.

[27] Philipp Koehn, Hieu Hoang, Alexandra Birch, and Chris et al. Callison-Burch. 2007. MOSES: Open Source Toolkit for Statistical Machine Translation. In *Association for Computational Linguistics*. 177–180.

[28] Taku Kudo. 2018. Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[29] Taku Kudo and John Richardson. 2018. SentencePiece: A Simple and Language independent Subword Tokenizer and Detokenizer for Neural Text Processing. In *Association for Computational Linguistics*.

[30] Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised Machine Translation using Monolingual Corpora Only. In *International Conference on Learning Representations*.

[31] Constantine Lignos, Daniel Cohen, Yen-Chieh Lien, Pratik Mehta, W. Bruce Croft, and Scott Miller. 2019. The Challenges of Optimizing Machine Translation for Low Resource Cross-Language Information Retrieval. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 3497–3502.

[32] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *CoRR abs/1604.00788* (2015).

[33] J. Scott MacCarley. 1999. Should We Translate the Documents or the Queries in Cross-Language Information Retrieval?. In *Proceedings of the* 37th *Annual Meeting*. 208–214.

[34] Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Analyzing Uncertainty in Neural Machine Translation. In *The* 35th *International Conference on Machine Learning*. 3956–3965.

[35] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Association for Computational Linguistics*.

[36] Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A Deep Reinforced Model for Abstractive Summarization. In *International Conference on Learning Representations*.

[37] Prajit Ramachandran, Peter J. Liu, and Quoc V. Le. 2018. Unsupervised Pretraining for Sequence to Sequence Learning. In *arXiv:1611.02683v2*.

[38] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the* 54th *Annual Meeting of the Association for Computational Linguistics*. 86–96.

[39] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In *The Conference on Neural Information Processing Systems*. 3104–3112.

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *The Conference on Neural Information Processing Systems*.

[41] Jianjun Zhang and Chengqing Zong. 2016. Exploiting Source-Side Monolingual Data in Neural Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1535–1545.

[42] Bing Zhao, Matthias Eck, and Stephan Vogel. 2004. Language Model Adaptation for Statistical Machine Translation with Structured Query Models. In *COLING*.

[43] Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer Learning for Low-Resource Neural Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1568–1575.

[44] Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On Using Monolingual Corpora in Neural Machine Translation. *CoRR abs/1503.03535* (2015).