Active Learning for Interactive Neural Machine Translation of Data Streams

Álvaro Peris and **Francisco Casacuberta**
Pattern Recognition and Human Language Technology Research Center
Universitat Politècnica de València, València, Spain
{lvapeab, fcn}@prhlt.upv.es

Abstract

We study the application of active learning techniques to the translation of unbounded data streams via interactive neural machine translation. The main idea is to select, from an unbounded stream of source sentences, those worth to be supervised by a human agent. The user will interactively translate those samples. Once validated, these data is useful for adapting the neural machine translation model.

We propose two novel methods for selecting the samples to be validated. We exploit the information from the attention mechanism of a neural machine translation system. Our experiments show that the inclusion of active learning techniques into this pipeline allows to reduce the effort required during the process, while increasing the quality of the translation system. Moreover, it enables to balance the human effort required for achieving a certain translation quality. Moreover, our neural system outperforms classical approaches by a large margin.

1 Introduction

The translation industry is a high-demand field. Large amounts of data must be translated on a regular basis. Machine translation (MT) techniques greatly boost the productivity of the translation agencies (Arenas, 2008). However, despite the recent advances achieved in this field, MT systems are still far to be perfect and make errors. The correction of such errors is usually done in a post-processing step, called post-editing. This requires a great effort, as it needs from expert human supervisors.

The requirements of the translation industry have increased in the last years. We live in a global world, in which large amounts of data must be periodically translated. This is the case of the European Parliament, whose proceedings must be regularly translated; or the Project Syndicate[1] platform, which translates editorials from newspapers to several languages. In these scenarios, the sentences to be translated can be seen as unbounded streams of data (Levenberg et al., 2010).

When dealing with such massive volumes of data, it is prohibitively expensive to manually revise all the translations. Therefore, it is mandatory to spare human effort, at the expense of some translation quality. Hence, when facing this situation, we have a twofold objective: on the one hand, we aim to obtain translations with the highest quality possible. On the other hand, we are constrained by the amount of human effort spent in the supervision and correction process of the translations proposed by an MT system.

The active learning (AL) framework is well-suited for these objectives. The application of AL techniques to MT involve to ask a human oracle to supervise a fraction of the incoming data (Bloodgood and Callison-Burch, 2010). Once the human has revised these samples, they are used for improving the MT system, via incremental learning. Therefore, a key element of AL is the so-called sampling strategy, which determines the sentences that should be corrected by the human.

Aiming to reduce the human effort required during post-editing, other alternative frameworks have been study. A successful one is the interactive-predictive machine translation (IMT) paradigm (Foster et al., 1997; Barrachina et al., 2009). In IMT, human and MT system jointly collaborate for obtaining high-quality translations, while reducing the human effort spent in this process.

In this work, we explore the application of NMT to the translation of unbounded data streams. We apply AL techniques for selecting the instances to

---

[1] www.project-syndicate.org

be revised by a human oracle. The correction process is done by means of an interactive-predictive NMT (INMT) system, which aims to reduce the human effort of this process. The supervised samples will be used for the NMT system to incrementally improve its models. To the best of our knowledge, this is the first work that introduces an INMT system into the scenario involving the translation of unbounded data. Our main contributions are:

- We study the application of AL on an INMT framework when dealing with large data streams. We introduce two sampling strategies for obtaining the most useful samples to be supervised by the human. We compare these techniques with other classical, well-performing strategies.

- We conduct extensive experiments, analyzing the different sampling strategies and studying the amount of effort required for obtaining a certain translation quality.

- The results show that AL succeeds at improving the translation pipeline. The translation systems featuring AL have better quality and require less human effort in the IMT process than static systems. Moreover, the application of the AL framework allows to obtain a balance between translation quality and effort required for achieving such quality. This balance can be easily tuned, according to the needs of the users.

- We open-source our code[2] and use publicly-available corpora, fostering further research on this area.

2 Related work

The translation of large data streams is a problem that has been thoroughly studied. Most works aim to continuously modify the MT system as more data become available. These modifications are usually performed in an incremental way (Levenberg et al., 2010; Denkowski et al., 2014; Turchi et al., 2017), learning from user post-edits. This incremental learning has also been applied to IMT, either to phrase-based statistical machine translation (SMT) systems (Nepveu et al., 2004; Ortiz-Martínez, 2016) or NMT (Peris and Casacuberta, 2018b).

The translation of large volumes of data is a scenario very appropriate for the AL framework (Cohn et al., 1994; Olsson, 2009; Settles, 2009). The application of AL to SMT has been studied for pool-based (Haffari et al., 2009; Bloodgood and Callison-Burch, 2010) and stream-based (González-Rubio et al., 2011) setups. Later works (González-Rubio et al., 2012; González-Rubio and Casacuberta, 2014), combined AL together with IMT, showing that AL can effectively reduce the human effort required for achieving a certain translation quality.

All these works were based on SMT systems. However, the recently introduced NMT paradigm (Sutskever et al., 2014; Bahdanau et al., 2015) has irrupted as the current state-of-the-art for MT (Bojar et al., 2017). Several works aimed at building more productive NMT systems. Related to our work, studies on interactive NMT systems (Knowles and Koehn, 2016; Peris et al., 2017; Hokamp and Liu, 2017) proved the efficacy of this framework. A body of work has been done aiming to build adaptive NMT systems, which continuously learn from human corrections (Turchi et al., 2017; Peris and Casacuberta, 2018b). Recently, Lam et al. (2018) applied AL techniques to an INMT system, for deciding whether the user should revise a partial hypothesis or not. However, to our knowledge, a study on the use of AL for NMT in a scenario of translation of unbounded data streams is still missing.

3 Neural machine translation

NMT is a particular case of sequence-to-sequence learning: given a sequence of words from the source language, the goal is to generate another sequence of words in the target language. This is usually done by means of an encoder–decoder architecture (Sutskever et al., 2014; Vaswani et al., 2017). In this work, we use a recurrent encoder–decoder system with long short-term memory (LSTM) units (Hochreiter and Schmidhuber, 1997) and an attention mechanism (Bahdanau et al., 2015).

Each element from the input sequence is projected into a continuous space by means of an embedding matrix. The sequence of embeddings is then processed by a bidirectional (Schuster and Paliwal, 1997) LSTM network, that concatenates the hidden states from forward and backward layers and produces a sequence of annotations.

The decoder is a conditional LSTM (cLSTM) network (Peris and Casacuberta, 2018b). A cLSTM network is composed of several LSTM transition blocks with an attention mechanism in between. We use two LSTM blocks.

The output of the decoder is combined together with the attended representation of the input sentence and with the word embedding of the word previously generated in a deep output layer (Pascanu et al., 2014). Finally, a softmax layer computes a probability distribution over the target language vocabulary.

The model is jointly trained by means of stochastic gradient descent (SGD) (Robbins and Monro, 1951), aiming to minimize the cross-entropy over a bilingual training corpus. SGD is usually applied to mini-batches of data; but it can be also applied sample-to-sample, allowing the training of the NMT system in an incremental way (Turchi et al., 2017).

For decoding, the model uses a beam search method (Sutskever et al., 2014) for obtaining the most probable target sentence $\hat{\mathbf{y}}$, given a source sentence $\mathbf{x}$:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}) \qquad (1)$$

3.1 Interactive machine translation

As previously discussed, MT systems are not perfect. Their outputs must be corrected by a human agent in a post-editing stage, in order to achieve high-quality translations.

The IMT framework constitutes a more efficient alternative to the regular post-editing. In a nutshell, IMT consists in an iterative process in which, at each iteration, the user introduces a correction to the system hypothesis. The system takes into account the correction and provides an alternative hypothesis, considering the feedback from the user.

In this work, we use a prefix-based IMT protocol: the user corrects the left-most wrong character of the hypothesis. With this action, the user has also validated a correct prefix. Then, the system must complete the provided prefix, generating a suitable suffix. Fig. 1 shows an example of the prefix-based IMT protocol.

More formally, the expression for computing the most probable suffix ($\hat{\mathbf{y}}_s$) is:

$$\hat{\mathbf{y}}_s = \arg\max_{\mathbf{y}_s} p(\mathbf{y}_s \mid \mathbf{x}, \mathbf{y}_p) \qquad (2)$$

Source (x):		They are lost forever .
Target (ŷ):		Ils sont perdus à jamais .
IT-0	MT	Ils sont perdus pour toujours .
IT-1	User	*Ils sont perdus* à̲ pour toujours .
	MT	*Ils sont perdus à* jamais .
IT-2	User	*Ils sont perdus à jamais .*

Figure 1: IMT session to translate a sentence from English to French. **IT-** is the number of iterations of the process. The **MT** row shows the MT hypothesis in the current iteration. In the **User** row is the feedback introduced by the user: the corrected character (boxed). We color in green the prefix that the user inherently validated with the character correction.

where $\mathbf{y}_p$ is the validated prefix provided by the user and $\mathbf{x}$ is the source sentence. Note that this expression is similar to Eq. (1). The difference is that now, the search space is the set of suffixes that complete $\mathbf{y}_p$.

For NMT systems, Eq. (2) is implemented as a beam search, constrained by the prefix provided by the user (Peris et al., 2017; Peris and Casacuberta, 2018b).

4 Active learning in machine translation

When dealing with potentially unbounded datasets, it becomes prohibitively expensive to manually supervise all the translations. Aiming to address this problem, in the AL framework, a sampling strategy selects a subset of sentences worth to be supervised by the user. Once corrected, the MT system adapts its models with these samples.

Therefore, the AL protocol applied to unbounded data streams is as follows (González-Rubio et al., 2012): first, we retrieve from the data stream $\mathcal{S}$ a block $\mathcal{B}$ of consecutive sentences, with the function getBlockFromStream($\mathcal{S}$). According to the sampling($\mathcal{B}, \varepsilon$) function, we select from $\mathcal{B}$ a subset $\mathcal{V}$ of ε instances, worth to be supervised by the user. See Section 5 for deeper insights on the sampling functions used in this work. These sampled sentences are interactively translated together with the user (Section 3.1). This process is done in the function INMT($\theta, \mathbf{x}, \mathbf{y}$). Once the user translates via INMT a source sentence $\mathbf{x}$, a correct translation $\hat{\mathbf{y}}$ is obtained. Then, we use the pair $(\mathbf{x}, \hat{\mathbf{y}})$ to retrain the parameters

Algorithm 1: Active learning for unbounded data streams with interactive neural machine translation.

input : θ (NMT model)
$\mathcal{S}$ (stream of source sentences)
ε (effort level desired)
auxiliar : $\mathcal{B}$ (block of source sentences)
$\mathcal{V} \subseteq \mathcal{B}$ (sentences to be supervised by the user)

1 **begin**
2   **repeat**
3        $\mathcal{B} = \mathtt{getBlockFromStream}(\mathcal{S})$;
4        $\mathcal{V} = \mathtt{sampling}(\mathcal{B}, \varepsilon)$;
5   **foreach** $\mathbf{x} \in \mathcal{B}$ **do**
6           $\mathbf{y} = \mathtt{translate}(\theta, \mathbf{x})$;
7   **if** $\mathbf{x} \in \mathcal{V}$ **then**
8              $\hat{\mathbf{y}} = \mathtt{INMT}(\theta, \mathbf{x}, \mathbf{y})$;
9              $\theta = \mathtt{update}(\theta, (\mathbf{x}, \hat{\mathbf{y}}))$;
10             $\mathtt{output}(\hat{\mathbf{y}})$;
11   **else**
12             $\mathtt{output}(\mathbf{y})$;
13   **end**
14   **end**
15   **until** $\mathcal{S} \neq \emptyset$;
16 **end**

θ from the NMT model, via SGD. This is done with the function $\mathtt{update}(\theta, (\mathbf{x}, \hat{\mathbf{y}}))$. Therefore, the NMT system is incrementally adapted with new data. The sentences considered unworthy to be supervised are automatically translated according to according Eq. (1), with the function $\mathtt{translate}(\theta, \mathbf{x})$. Once we finish the translation of the current block $\mathcal{B}$, we start the process again. Algorithm 1 details the full procedure.

5   Sentence sampling strategies

One of the key elements of AL is to have a meaningful strategy for obtaining the most useful samples to be supervised by the human agent. This requires an evaluation of the informativeness of unlabeled samples. The sampling strategies used in this work belong to two major frameworks: uncertainty sampling (Lewis and Catlett, 1994) and query-by-committee (Seung et al., 1992).

As baseline, we use a random sampling strategy: sentences are randomly selected from the data stream $\mathcal{S}$. Although simple, this strategy usually works well in practice. In the rest of this section, we describe the sampling strategies used in

this work.

5.1   Uncertainty sampling

The idea behind this family of methods is to select those instances for which the model has the least confidence to be properly translated. Therefore, all techniques compute, for each sample, an uncertainty score. The selected sentences will be those with the highest scores.

Quality estimation sampling

A common and effective way for measuring the uncertainty of a MT system is to use confidence estimation (Gandrabur and Foster, 2003; Blatz et al., 2004; Ueffing and Ney, 2007). The idea is to estimate the quality of a translation according to confidence scores of the words.

More specifically, given a source sentence $\mathbf{x} = x_1, \ldots, x_J$ and a translation hypothesis $\mathbf{y} = y_1, \ldots, y_I$, a word confidence score (C_w) as computed as (Ueffing and Ney, 2005):

$$C_w(\mathbf{x}, y_i) = \max_{0 \leq j \leq J} p(y_i | x_j) \qquad (3)$$

where $p(y_i | x_j)$ is the alignment probability of y_i and x_j, given by an IBM Model 2 (Brown et al., 1993). x_0 denotes the empty source word. The choice of the IBM Model 2 is twofold: on the one hand, it is a very fast method, which only requires to query in a dictionary. We are in an interactive framework, therefore speed becomes a crucial requirement. On the other hand, its performance is close to more complex methods (Blatz et al., 2004; Dyer et al., 2013).

Following González-Rubio et al. (2012), the uncertainty score for the quality estimation sampling is defined as:

$$C_{\mathrm{qe}}(\mathbf{x}, \mathbf{y}) = 1 - \frac{|\{y_i \in \mathbf{y} | C_w(\mathbf{x}, y_i) > \tau_w\}|}{|\mathbf{y}|} \quad (4)$$

where τ_w is a word confidence threshold, adjusted according to a development corpus. $|\cdot|$ denotes the size of a sequence or set.

Coverage sampling

One of the main issues suffered by NMT systems is the lack of coverage: the NMT system may not translate all words from a source sentence. This results in over-translation or under-translation problems (Tu et al., 2016).

We propose to use the translation coverage as a measure of the uncertainty suffered by the NMT

system when translating a sentence. Therefore, we modify the coverage penalty proposed by Wu et al. (2016), for obtaining a coverage-based uncertainty score:

$$C_{\text{cov}}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{j=1}^{|\mathbf{x}|} \log \left(\min(\sum_{i=1}^{|\mathbf{y}|} \alpha_{i,j}, 1) \right)}{|\mathbf{x}|}$$

(5)

where $\alpha_{i,j}$ is attention probability of the i-th target word and the j-th source word.

Attention distraction sampling

When generating a target word, an attentional NMT system should attend on meaningful parts of the source sentence. If the system is translating an uncertain sample, its attention mechanism will be *distracted*. That means, dispersed throughout the source sequence. A sample with a great distraction will feature an attention probability distribution with heavy tails (e.g. a uniform distribution). Therefore, for the attention distraction sampling strategy, the sentences to select will be those with highest attention distraction.

For computing a distraction score, we compute the kurtosis of the weights given by the attention model for each target word y_i:

$$\text{Kurt}(y_i) = \frac{\frac{1}{|\mathbf{x}|} \sum_{j=1}^{|\mathbf{x}|} (\alpha_{i,j} - \frac{1}{|\mathbf{x}|})^4}{\left(\frac{1}{|\mathbf{x}|} \sum_{j=1}^{|\mathbf{x}|} (\alpha_{i,j} - \frac{1}{|\mathbf{x}|})^2 \right)^2}$$

(6)

being, as above, $\alpha_{i,j}$ the weight assigned by the attention model to the j-th source word when decoding the i-th target word. Note that, by construction of the attention model, $\frac{1}{|\mathbf{x}|}$ is equivalent to the mean of the attention weights of the word y_i.

Since we want to obtain samples with heavy tails, we average the minus kurtosis values for all words in the target sentence, obtaining the attention distraction score C_{ad}:

$$C_{\text{ad}}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{|\mathbf{y}|} -\text{Kurt}(y_i)}{|\mathbf{y}|}$$

(7)

5.2 Query-by-committee

This framework maintains a committee of models, each one able to vote for the sentences to be selected. The query-by-committee (QBC) method selects the samples with the largest disagreement among the members of the committee. The level of disagreement of a sample $\mathbf{x}$ measured according to the vote-entropy function (Dagan and Engelson, 1995):

$$C_{\text{qbc}}(\mathbf{x}) = -\frac{\#V(\mathbf{x})}{|\mathcal{C}|} + \log \frac{\#V(\mathbf{x})}{|\mathcal{C}|}$$

(8)

where $\#V(\mathbf{x})$ is the number of members of the committee that voted $\mathbf{x}$ to be worth to be supervised and $|\mathcal{C}|$ is the number of members of the committee. If $\#V(\mathbf{x})$ is zero, we set the value of $C_{\text{qbc}}(\mathbf{x})$ to $-\infty$.

Our committee was composed by the four uncertainty sampling strategies, namely quality estimation, coverage, attention distraction and random sampling. The inclusion of the latter into the committee can be seen as a way of introducing some noise, aiming to prevent overfitting.

6 Experimental framework

In order to assess the effectiveness of AL for INMT, we conducted a similar experimentation than the latter works in AL for IMT (González-Rubio and Casacuberta, 2014): we started from a NMT system trained on a general corpus and followed Algorithm 1. This means that the sampling strategy selected those instances to be supervised by the human agent, who interactively translated them. Next, the NMT system was updated in an incremental way with the selected samples.

Due to the prohibitive cost that an experimentation with real users conveys, in our experiments, the users were simulated. We used the references from our corpus as the sentences the users would like to obtain.

6.1 Evaluation

An IMT scenario with AL requires to assess two different criteria: translation quality of the system and human effort spent during the process.

For evaluating the quality of the translations, we used the BLEU (bilingual evaluation understudy) (Papineni et al., 2002) score. BLEU computes an average mean of the precision of the n-grams (up to order 4) from the hypothesis that appear in the reference sentence. It also has a brevity penalty for short translations.

For estimating the human effort, we simulated the actions that the human user would perform when using the IMT system. Therefore, at each iteration the user must search in the hypothesis the next error, and position the mouse pointer on it.

Once the pointer is positioned, the user would introduce the correct character. These actions correspond to a *mouse-action* and a *keystroke*, respectively.

Therefore, we use a commonly-used metric that accounts for both types of interaction: the keystroke mouse-action ratio (KSMR) (Barrachina et al., 2009). It is defined as the number of keystrokes plus the number of mouse-actions required for obtaining the desired sentence, divided by the number of characters of such sentence. We add a final mouse-action, accounting for action of accepting the translation hypothesis. Although keystrokes and mouse-actions are different and require a different amount of effort (Macklovitch et al., 2005), KSMR makes an approximation and assumes that both actions require a similar effort.

6.2 Corpora

To ensure a fair comparison with the latter works of AL applied to IMT (González-Rubio and Casacuberta, 2014), we used the same datasets: our training data was the Europarl corpus (Koehn, 2005), with the development set provided at the 2006 workshop on machine translation (Koehn and Monz, 2006). As test set, we used the News Commentary corpus (Callison-Burch et al., 2007). This test set is suitable to our problem at hand because i. it contains data from different domains (politics, economics and science), which represent challenging out-of-domain samples, but account for a real-life situation in a translation agency; and ii. it is large enough to properly simulate long-term evolution of unbounded data streams. All data are publicly available. We conducted the experimentation in the Spanish to English language direction. Table 1 shows the main figures of our data.

Table 1: Corpora main figures, in terms of number of sentences ($|S|$), number of running words ($|W|$) and vocabulary size ($|V|$). k and M stand for thousands and millions of elements, respectively.

Corpus	Usage		$|S|$	$|W|$	$|V|$
Europarl	Train	En	2M	46M	106k
		Es		48M	160k
	Dev.	En	2k	58k	6.1k
		Es		61k	7.7k
News Commentary	Test	En	51k	1.2M	35k
		Es		1.5M	49k

6.3 NMT systems and AL setup

Our NMT system was built using NMT-Keras (Peris and Casacuberta, 2018a) and featured a bidirectional LSTM encoder and a decoder with cLSTM units. Following Britz et al. (2017), we set the dimension of the LSTM, embeddings and attention model to 512. We applied batch normalizing transform (Ioffe and Szegedy, 2015) and Gaussian noise during training (Graves, 2011). The L_2 norm of the gradients was clipped to 5, for avoiding the exploiting gradient effect (Pascanu et al., 2012). We applied joint byte pair encoding (BPE) (Sennrich et al., 2016) to all corpora. For training the system, we used Adam (Kingma and Ba, 2014), with a learning rate of 0.0002 and a batch size of 50. We early-stopped the training according to the BLEU on our development set. For decoding, we used a beam of 6.

We incrementally update the system (Line 9 in Algorithm 1), with vanilla SGD, with a learning rate of 0.0005. We chose this configuration according to an exploration on the validation set.

The rest of hyperparameters were set according to previous works. The blocks retrieved from the data stream contained 500 samples (according to González-Rubio et al. (2012), the performance is similar regardless the block size). For the quality estimation method, the IBM Model 2 was obtained with `fast_align` (Dyer et al., 2013) and τ_w was set to 0.4 (González-Rubio et al., 2010).

7 Results and discussion

A system with AL involves two main facets to evaluate: the improvement on the quality of the system and the amount of human effort required for achieving such quality. In this section, we compare and study our AL framework for all our sampling strategies: quality estimation sampling (QES), coverage sampling (CovS), attention distraction sampling (ADS), random sampling (RS) and query-by-committee (QBC).

7.1 Active learning evaluation

First, we evaluated the effectiveness of the application of AL in the NMT system, in terms of translation quality. Fig. 2 shows the BLEU of the initial hypotheses proposed by the NMT system (Line 6 in Algorithm 1), as a function of the percentage of sentences supervised by the user (ε in Algorithm 1). That means, the percentage of sentences used to adapt the system. The BLEU of a static

system without AL was 34.6. Applying AL, we obtained improvements up to 4.1 points of BLEU.
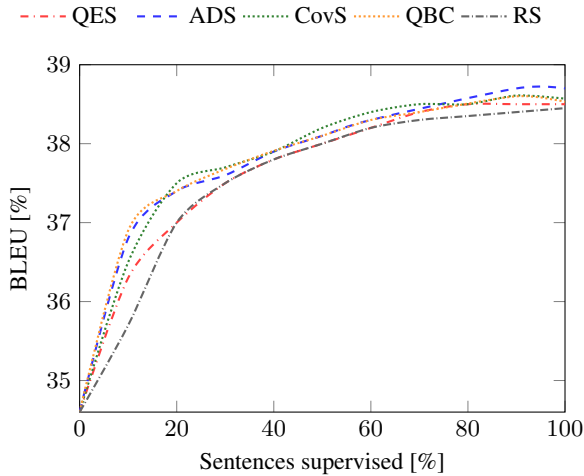

Figure 2: BLEU of the initial hypotheses proposed by the the NMT system as a function of the amount of data used to adapt it. The percentage of sentences supervised refers to the value of ε with respect to the block size.

As expected, the addition of the new knowledge had a larger impact when applied to a non-adapted system. Once the system becomes more specialized, a larger amount of data was required to further improve.

The sampling strategies helped the system to learn faster. Taking RS as a baseline, the learning curves of the other techniques were better, especially when using few (up to a 30%) data for fine-tuning the system. The strategies that achieved a fastest adaptation were those involving the attention mechanism (ADS, CovS and QBC). This indicates that the system is learning from the most useful data. The QES and RS required more supervised data for achieving the comparable BLEU results. When supervising high percentages of the data, we observed BLEU differences. This is due to the ordering in which the selected sentences were presented to the learner. The sampling strategies performed a sort of curriculum learning (Bengio et al., 2009).

7.2 Introducing the human into the loop

From point of view a user, it is important to assess not only the quality of the MT system, but also the effort spent to obtain such quality. Fig. 3 relates both, showing the amount of effort required for obtaining a certain translation quality. We compared the results of system with AL against the same NMT system without AL and with two other SMT systems, with and without AL, from González-Rubio and Casacuberta (2014).

Results in Fig. 3 show consistent positive results of the AL framework. In all cases, AL reduced the human effort required for achieving a certain translation quality. Compared to a static NMT system, approximately a 25% of the human effort can be spent using AL techniques.
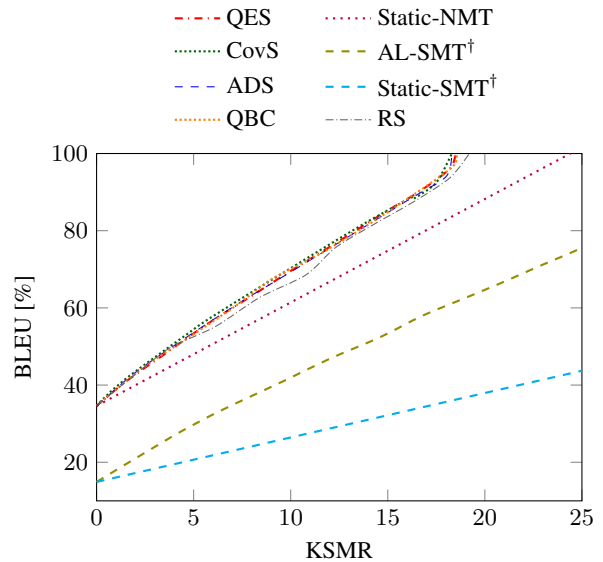

Figure 3: Translation quality (BLEU) as a function of the human effort (KSMR) required. Static-NMT relates to the same NMT system without AL. [†] denotes systems from González-Rubio and Casacuberta (2014): Static-SMT is a SMT system without AL and AL-SMT is the coverage augmentation SMT system.

Regarding the different sampling strategies, all of them behaviored similarly. They provided consistent and stable improvements, regardless the level of effort desired (ε). This indicates that, although the BLEU of the system may vary (Fig. 2), this had small impact on the effort required for correcting the samples. All sampling strategies outperformed the random baseline, which had a more unstable behavior.

Compared to classical SMT systems, NMT performed surprisingly well. Even the NMT system without AL largely outperformed the best AL-SMT system. This is due to several reasons: on the one hand, the initial NMT system was much better than the original SMT system (34.6 vs. 14.9 BLEU points). Part of this large difference were

presumably due to the BPE used in NMT: the data stream contained sentences from different domains, but they can be effectively encoded into known sequences via BPE. The SMT system was unable to handle well such unseen sentences. On the other hand, INMT systems usually respond much better to the human feedback than interactive SMT systems (Knowles and Koehn, 2016; Peris et al., 2017). Therefore, the differences between SMT and NMT were enlarged even more.

Finally, it should be noted that all our sampling strategies can be computed speedily. They involve analysis of the NMT attention weights, which are computed as a byproduct of the decoding process; or queries to a dictionary (in the case of QES). The update of NMT system is also fast, taking approximately 0.1 seconds. This makes AL suitable for a real-time scenario.

8 Conclusions and future work

We studied the application of AL methods to INMT systems. The idea was to supervise the most useful samples from a potentially unbounded data stream, while automatically translating the rest of samples. We developed two novel sampling strategies, able to outperform other well-established methods, such as QES, in terms of translation quality of the final system.

We evaluated the capabilities and usefulness of the AL framework by simulating real-life scenario, involving the aforementioned large data streams. AL was able to enhance the performance of the NMT system in terms of BLEU. Moreover, we obtained consistent reductions of approximately a 25% of the effort required for reaching a desired translation quality. Finally, it is worth noting that NMT outperformed classical SMT systems by a large margin.

We want to explore several lines of work in a future. First, we intend to apply our method to other datasets, involving linguistically diverse language pairs and low-resource scenarios, in order to observe whether the results obtained in this work hold. We also aim to devise more effective sampling strategies. To take into account the cognitive effort or time required for interactively translating a sentence seem promising objective functions. Moreover, these sampling strategies can be used as a data selection technique. It would be interesting to assess their performance on this task. We also want to study the addition of reinforcement or bandit learning into our framework. Recent works (Nguyen et al., 2017; Lam et al., 2018) already showed the usefulness of these learning paradigms, which are orthogonal to our work. Finally, we intend to assess the effectiveness of our proposals with real users in a near future.

References

Ana Guerberof Arenas. 2008. Productivity and quality in the post-editing of outputs from translation memories and machine translation. *Localisation Focus*, 7(1):11–21.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*.

Sergio Barrachina, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, and Juan-Miguel Vilar. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *Proceedings of the international conference on Computational Linguistics*, pages 315–321.

Michael Bloodgood and Chris Callison-Burch. 2010. Bucking the trend: Large-scale cost-focused active learning for statistical machine translation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 854–864.

Ondřej Bojar, Christian Buck, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, and Julia Kreutzer, editors. 2017. *Proceedings of the Second Conference on Machine Translation*.

Denny Britz, Anna Goldie, Thang Luong, and Quoc Le. 2017. Massive exploration of neural machine translation architectures. *arXiv:1703.03906*.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2007. (Meta-) evaluation of machine translation. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 136–158.

David Cohn, Les Atlas, and Richard Ladner. 1994. Improving generalization with active learning. *Machine learning*, 15(2):201–221.

Ido Dagan and Sean P Engelson. 1995. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157.

Michael Denkowski, Chris Dyer, and Alon Lavie. 2014. Learning from post-editing: Online model adaptation for statistical machine translation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 395–404.

Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of IBM Model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648.

George Foster, Pierre Isabelle, and Pierre Plamondon. 1997. Target-text mediated interactive machine translation. *Machine Translation*, 12:175–194.

Simona Gandrabur and George Foster. 2003. Confidence estimation for text prediction. In *Proceedings of the Conference on Computational Natural Language Learning*, pages 315–321.

Jesús González-Rubio and Francisco Casacuberta. 2014. Cost-sensitive active learning for computer-assisted translation. *Pattern Recognition Letters*, 37:124–134.

Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2010. Balancing user effort and translation error in interactive machine translation via confidence measures. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 173–177.

Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2011. An active learning scenario for interactive machine translation. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 197–200.

Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2012. Active learning for interactive machine translation. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 245–254.

Alex Graves. 2011. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*, pages 2348–2356.

Gholamreza Haffari, Maxim Roy, and Anoop Sarkar. 2009. Active learning for statistical phrase-based machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 415–423.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. *arXiv:1704.07138*.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Rebecca Knowles and Philipp Koehn. 2016. Neural interactive translation prediction. In *Proceedings of the Association for Machine Translation in the Americas*, pages 107–120.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the Machine Translation Summit*, pages 79–86.

Philipp Koehn and Christof Monz, editors. 2006. *Proceedings on the Workshop on Statistical Machine Translation*. Association for Computational Linguistics.

Tsz Kin Lam, Julia Kreutzer, and Stefan Riezler. 2018. A reinforcement learning approach to interactive-predictive neural machine translation. In *Proceedings of the European Association for Machine Translation conference*, pages 169–178.

Abby Levenberg, Chris Callison-Burch, and Miles Osborne. 2010. Stream-based translation models for statistical machine translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 394–402.

David D Lewis and Jason Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *Machine Learning Proceedings 1994*, pages 148–156.

Elliot Macklovitch, Nam-Trung Nguyen, and Roberto Silva. 2005. User evaluation report. Technical report. Transtype2 (ISR-2001-32091).

Laurent Nepveu, Guy Lapalme, Philippe Langlais, and George Foster. 2004. Adaptive language and translation models for interactive machine translation. In *Proceedings of the Conference on Empirical Method in Natural Language Processing*, pages 190–197.

Khanh Nguyen, Hal Daumé III, and Jordan Boyd-Graber. 2017. Reinforcement learning for bandit neural machine translation with simulated human feedback. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1464–1474.

Fredrik Olsson. 2009. A literature survey of active machine learning in the context of natural language processing. Technical report.

Daniel Ortiz-Martínez. 2016. Online learning for statistical machine translation. *Computational Linguistics*, 42(1):121–161.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. How to construct deep recurrent neural networks. *arXiv:1312.6026*.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2012. On the difficulty of training recurrent neural networks. *arXiv:1211.5063*.

Álvaro Peris and Francisco Casacuberta. 2018a. NMT-Keras: a very flexible toolkit with a focus on interactive NMT and online learning. *The Prague Bulletin of Mathematical Linguistics*, 111:113–124.

Álvaro Peris and Francisco Casacuberta. 2018b. Online learning for effort reduction in interactive neural machine translation. *arXiv:1802.03594*.

Álvaro Peris, Miguel Domingo, and Francisco Casacuberta. 2017. Interactive neural machine translation. *Computer Speech & Language*, 45:201–220.

Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 86–96.

B. Settles. 2009. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison.

H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. 1992. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the Advances in Neural Information Processing Systems*, volume 27, pages 3104–3112.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 76–85.

Marco Turchi, Matteo Negri, M Amin Farajian, and Marcello Federico. 2017. Continuous learning from human post-edits for neural machine translation. *The Prague Bulletin of Mathematical Linguistics*, 108(1):233–244.

Nicola Ueffing and Hermann Ney. 2005. Application of word-level confidence measures in interactive statistical machine translation. In *Proceedings of the European Association for Machine Translation conference*, pages 262–270.

Nicola Ueffing and Hermann Ney. 2007. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33:9–40.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv:1706.03762*.

Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Ł. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv:1609.08144*.