Interactive Neural Machine Translation

Álvaro Peris*, Miguel Domingo, Francisco Casacuberta

Pattern Recognition and Human Language Technology Research Center,
Universitat Politècnica de València,
Camino de Vera s/n, 46022 Valencia, SPAIN

Abstract

Despite the promising results achieved in last years by statistical machine translation, and more precisely, by the neural machine translation systems, this technology is still not error-free. The outputs of a machine translation system must be corrected by a human agent in a post-editing phase. Interactive protocols foster a human–computer collaboration, in order to increase productivity. In this work, we integrate the neural machine translation into the interactive machine translation framework. Moreover, we propose new interactivity protocols, in order to provide the user an enhanced experience and a higher productivity. Results obtained over a simulated benchmark show that interactive neural systems can significantly improve the classical phrase-based approach in an interactive-predictive machine translation scenario.

Keywords: Neural machine translation; Interactive-predictive machine translation; Recurrent neural networks

1. Introduction

The statistical framework has allowed a breakthrough in machine translation (MT) and new systems provide admissible results for many tasks. However, in other scopes the quality of fully-automated systems is insufficient. In such cases, MT is used to obtain translation hypotheses, which must be supervised and corrected by a human agent in a post-editing (PE) stage. This working method is more productive than a completely manual translation, since the translator starts from an initial hypothesis that must be corrected. Nevertheless, this is a decoupled strategy in which computer and human agent work independently. Higher efficiency rates can be reached if human and system collaborate on a joint strategy. Seeking for this human–computer collaboration, Foster et al. (1997) introduced the so-called interactive-predictive MT (IMT), further developed by Alabau et al. (2013), Barrachina et al. (2009), Bender et al. (2005), Langlais and Lapalme (2002) and Macklovitch (2006).

This approach consists in an iterative prediction–correction process: each time the user corrects a word, the system reacts offering a new translation hypothesis, expected to be better than the previous one. In the basic IMT proposal, the user was constrained to follow a left-to-right protocol. Always was corrected the left-most wrong word from a translation hypothesis. This word, together with the previous ones, formed a *validated prefix*. At each iteration, the user validated a larger prefix and the system produced an appropriate suffix for completing the translation.

IMT evolved during the years, introducing advances related to the generation of the new suffix (Azadi and Khadivi, 2015; Cai et al., 2013; Green et al., 2014b; Koehn et al., 2014; Ortiz-Martínez, 2011), and the possibility of suggesting more than one suffix (Koehn, 2010; Torregrosa et al., 2014). Other novelties came from profiting the use of the mouse, validating a prefix and suggesting a new suffix each time the user

*Corresponding author: Phone: (34) 96 387 70 69
Email addresses: lvapeab@prhlt.upv.es (Álvaro Peris), midobal@prhlt.upv.es (Miguel Domingo), fcn@prhlt.upv.es (Francisco Casacuberta)

clicked into a position to type a word correction (Sanchis-Trilles et al., 2008). The addition of confidence measures aided the user to validate correct prefixes (González-Rubio et al., 2010a,b). The use of online learning techniques was also studied, aiming to improve the system with the user feedback (Mathur et al., 2014; Nepveu et al., 2004; Ortiz-Martínez, 2016). Related to this, González-Rubio et al. (2012) explored the active learning protocol in an interactive post-edition stage. An interactive approach was also developed for hierarchical translation models (González-Rubio et al., 2013). Multimodal interaction integrated handwriting recognition/speech recognition into the IMT environment (Alabau et al., 2011, 2014). Green et al. (2014a) investigated the interactive use of translation memories. Nonetheless, the core of the user protocol remained the same in all these cited works. Recent works (González-Rubio et al., 2016) strove to overcome the prefix-based approach. One of the interactive protocols proposed in our work, relies on these latter ideas of breaking down the prefix constraint.

The prefix-based protocol suffered from three main issues: first, it was quite restrictive. The human translator was forced to always follow the left-to-right validation direction. This could be unnatural for the users or even inadequate in many cases. Second, the IMT system could produce worse suffixes, which also should be corrected by the user. Apart from increasing the human effort of the process, this introduced an annoying behaviour: the user had to correct words that were right in previous iterations, leading to user exasperation. The third issue was the computational cost of the (prefix-constrained) search for alternative hypotheses, which prevented the use of regular decoders. The increase of the computational power has alleviated this problem, allowing the use of more complex models and search strategies, in order to reach real-time generation of successive hypotheses.

Pursuing to overcome both first problems, in this work we propose an alternative protocol: when a hypothesis is generated, the user can select correct word sequences, called *segments*, from all over the sentence. These segments are considered to be valid and will remain in future iterations. The user can also correct wrong words, as in the classical approach. The system offers then an alternative hypothesis, that takes into account the corrected word together with the validated segments. Thus, correct parts of the hypothesis are kept during successive interactions, offering a more comfortable user experience and an increase in the productivity.

Up to now, the IMT approaches were based on discrete representations of words and sentences. Nevertheless, in the last years, continuous representations of words and sentences have gained much the attention of the natural language processing community. Distributed representations are richer than classical ones, yielding encouraging results. Although neural models were already applied to MT long ago (Castaño and Casacuberta, 1997), they finally took off recently and its use has dramatically increased. Bengio et al. (2003) proposed to project words into a distributed space and estimate the probabilities of a language model in such space. From here, continuous models have been used profusely in a wide range of tasks like language modelling (Mikolov et al., 2010; Schwenk, 2007; Sundermeyer et al., 2012), handwritten text recognition (Graves et al., 2009) or automatic speech recognition (Graves et al., 2013). In the MT field, neural models have been successfully introduced into the current statistical machine translation (SMT) pipeline, both in the phrase-based and hierarchical approaches (Devlin et al., 2014; Sundermeyer et al., 2014).

In addition to this, a neural approach to MT has been recently proposed (Cho et al., 2014; Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014). Neural machine translation (NMT) has emerged as one of the most promising technologies to tackle the MT problem. It is based on the use of neural networks for building end-to-end systems. The translation problem is addressed by a single, large neural network, which reads an input sentence and directly generates its translation. This is opposed to classical approaches to MT (e.g. Koehn et al. (2003)), made up of multiple decoupled models. Most architectures are based on recurrent neural networks (RNN). In order to properly deal with long-term relationships, RNNs use gated units, such as long short-term memory (LSTM) units (Hochreiter and Schmidhuber, 1997) or gated recurrent units (GRU) (Cho et al., 2014).

There has been a significant effort for improving the NMT model. Thus, attention mechanisms were included to the model (Bahdanau et al., 2015; Luong et al., 2015b), allowing the model to focus on different parts of the input sentence. The out-of-vocabulary problem was tackled by Jean et al. (2015), Luong et al. (2015a) and Sennrich et al. (2016). Jean et al. (2015) also investigated the use of large target vocabularies. Gulcehre et al. (2015) included additional monolingual resources into the system. NMT at character-level

has also obtained promising results (Chung et al., 2016; Costa-Jussà and Fonollosa, 2016; Ling et al., 2015).

Furthermore, the same neural framework has also been recently applied in different tasks, such as image (Vinyals et al., 2015b; Xu et al., 2015) and video captioning (Peris et al., 2016; Yao et al., 2015), image generation (Gregor et al., 2015) or syntactic constituency parsing (Vinyals et al., 2015a).

NMT systems implement fairly simple decoders, especially compared to the classical PB ones. Hence, to modify the decoder for integrating interactive mechanisms is easier than in the PB (Sanchis-Trilles et al., 2014) or hierarchical MT approaches (González-Rubio et al., 2013).

In this work, we explore the integration of the NMT technology into the interactive framework. To the best of our knowledge, this is the first work that delves into the combination of the interactive paradigm together with neural systems. Our main contributions are the following:

1. We adapt the NMT paradigm to fit into the classical prefix-based interactive MT.
2. We implement an alternative interaction protocol that aims to offer the user more freedom and a more comfortable and more productive experience. We formulate and implement strategies for applying the NMT technology within the extended protocol.
3. We conduct a wide experimentation, tackling the different translation tasks usually employed in the literature. Such tasks feature different language pairs, domains and complexities. Results show significant improvements over the traditional PB systems.

This manuscript is structured as follows: after this introduction, Section 2 is devoted to the main concepts and statistical formalization of IMT. NMT is presented in Section 3. In Section 4, we deploy the integration of NMT systems into the IMT protocols. The experimental framework is set up in Section 5. In Section 6, we show and discuss the experimental results obtained. Finally, conclusions are drawn in Section 7.

2. Interactive machine translation

IMT arose as an alternative to the classical PE stage, in which a human agent supervised and corrected the output of an MT system in a decoupled manner. Under the interactive paradigm, the PE stage shifts to an iterative human–computer collaboration process: each time the user makes a correction, the system suggests a new translation according to the feedback introduced by the user. IMT strives hence, for combining the efficiency of an MT engine with the knowledge of a human translator.

Classical IMT proposals (Alabau et al., 2013; Barrachina et al., 2009) are based on the statistical formalization of the MT problem. The goal of SMT (Brown et al., 1993) is to find the best translation $\hat{y}_1^{\hat{I}} = \hat{y}_1, \ldots, \hat{y}_{\hat{I}}$ of length $\hat{I}$, given a source sentence $x_1^J = x_1, \ldots, x_J$ of length J:

$$\hat{y}_1^{\hat{I}} = \arg\max_{I, y_1^I} \Pr(y_1^I \mid x_1^J) \tag{1}$$

During several years, the state of the art in SMT were PB systems, based on bilingual phrases as translation units (Koehn et al., 2003; Zens et al., 2002). PB models combine different features by means of a log-linear model (Och and Ney, 2002). SMT based on stochastic finite state models also played a significant role in previous IMT works (Barrachina et al., 2009); but the prevailing SMT technology were PB systems (Koehn et al., 2007). Nevertheless, in the last few years, this trend is shifting in favour of neural approaches (Bojar et al., 2016). To the best of our knowledge, the integration of interactive protocols in the NMT approach remains unexplored.

2.1. Prefix-based interactive machine translation

In the classical prefix-based IMT protocol, computer and human collaborate to translate a source sentence x_1^J. This collaboration starts with the MT system proposing an initial translation $y_1^I = y_1, \ldots, y_I$. The user then searches, from the left to the right, the first wrong word y_t from the translation and corrects it. This provides the system a feedback signal with the form $f = \hat{y}_i$, where $\hat{y}_i$ is the corrected word at the position i in the target hypothesis. This feedback signal is double-folded: it states that the i-th target word must be $\hat{y}_i$,

but it also validates the hypothesis prefix y_1^{i-1}. Taking this into account, we can rewrite f as $f = \hat{y}_1^i$, where $\hat{y}_1^i$ is the validated prefix together with the corrected word. At the next iteration, the system generates the best suffix $y'^{I'}_{i+1}$ to build a new translation $y'^{I'}_1 = \hat{y}_1^i \; y'^{I'}_{i+1}$. This process is repeated until the user accepts the complete hypothesis of the system. Fig. 1 represents this protocol.

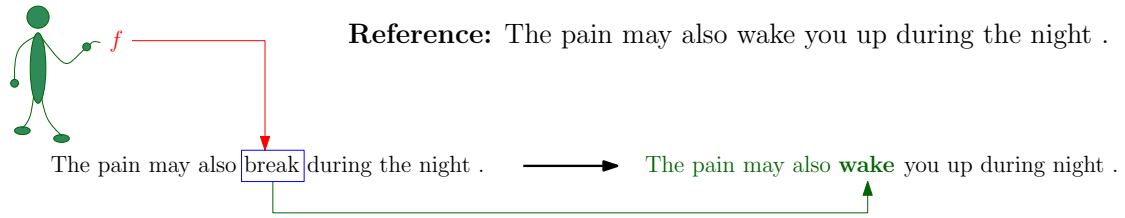

Figure 1: Single iteration of prefix-based IMT. The user wants to translate the French sentence "La doleur peut également vous réveiller pendant la nuit ." into English. The user corrects the first wrong word from the hypothesis provided by the system, introducing the word "wake" at position 5. Next, the system generates a new hypothesis, that contains the validated prefix together with the corrected word. Note that, although the system generates a partially correct suffix, in this new hypothesis it is also introduced a new error ("during night" instead of "during the night"). This behaviour is intended to be solved with the segment-based approach.

The suffix generation is formalized as follows (Barrachina et al., 2009):

$$y'^{I'}_{i+1} = \arg\max_{I, y^I_{i+1}} \Pr(y^I_{i+1} \mid x_1^J, \hat{y}_1^i) \tag{2}$$

which can be straightforwardly rewritten as:

$$y'^{I'}_{i+1} = \arg\max_{I, y^I_{i+1}} \Pr(\hat{y}_1^i, y^I_{i+1} \mid x_1^J) \tag{3}$$

This equation is very similar to Eq. (1). Therefore, at each iteration, the process consists of a regular search in the translations space, but constrained by the prefix $\hat{y}_1^i$.

2.2. Segment-based interactive machine translation

In the segment-based IMT protocol proposed in this work, collaboration between computer and human is extended. Now, besides correcting a wrong word, the user can validate segments (word sequences) to be kept in future iterations. As before, the process starts with the MT system proposing an initial translation $y_1^I = y_1, \ldots, y_I$. The user then validates all correct segments from y_1^I and introduces a word correction. These actions become a feedback signal $f_1^N = f_1, \ldots, f_N$, where $f_1, \ldots, f_N$ is a sequence of the N non-overlapping, validated segments, including a one-word segment with the word the user has corrected. At the next iteration, the system generates a sequence of *non-validated segments* $\tilde{g}_1^N = \tilde{g}_1, \ldots, \tilde{g}_N$ that fills f_1^N to conform a new translation $y' = f_1, \tilde{g}_1, \ldots, f_N, \tilde{g}_N$. Once again, this process is repeated until the user accepts the complete suggestion of the system. Fig. 2 represents the segment-based interactive protocol.

In our statistical framework, the best translation segments are obtained as:

$$\tilde{g}_1^N = \arg\max_{g_1^N} \Pr(g_1^N \mid x_1^J, f_1^N) \tag{4}$$

which can be rewritten as:

$$\tilde{g}_1^N = \arg\max_{g_1^N} \Pr(f_1, g_1, \ldots, f_N, g_N \mid x_1^J) \tag{5}$$

This last equation is very similar to the classical prefix-based IMT equation (Eq. (3)). The search process in Eq. (3) is limited to the space of suffixes, constrained by $\hat{y}_1^i$. In Eq. (5) the search space is all possible substrings of the translations of x_1^J, constrained by the sequence of segments $f_1, \ldots, f_N$.

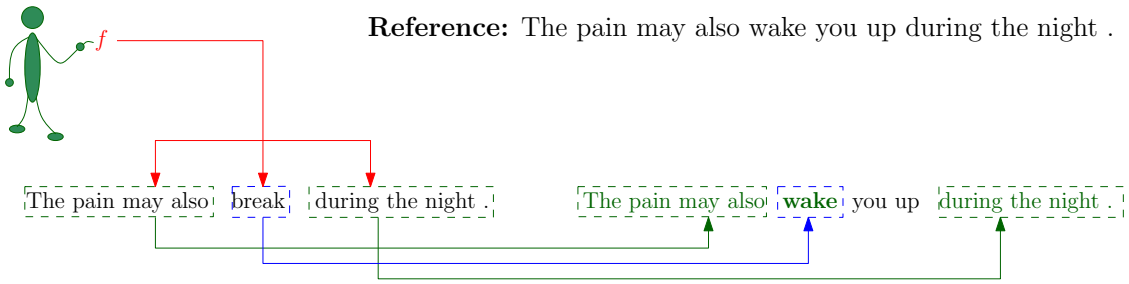

Figure 2: Segment-based IMT iteration for the same example than in Fig. 1. In this case, the user validates two segments and introduces a word correction. The system generates a new hypothesis that contains the word correction and keeps the validated segments. The user feedback is f = "The pain may also", "wake", "during the night .". The reaction of the system is to generate the sequence of non-validated segments $\tilde{g} = \lambda$, "you up", λ; being λ the empty string. The hypothesis offered by the system consists in the combination of the validated and non-validated segments.

3. Neural machine translation

Most NMT systems rely on an RNN encoder-decoder framework: at the encoding process, a source sentence is projected into a distributed representation. Given this representation, the decoder generates, word by word, the corresponding translated sentence. From Eq. (1), NMT aim to directly model the conditional translation probability:

$$\Pr(y_1^I|x_1^J) = \prod_{i=1}^{I} \Pr(y_i \mid y_1^{i-1}, x_1^J) \tag{6}$$

In this work, we followed the architectural choices made by Bahdanau et al. (2015). Refer to this work for a more in-depth review of the system. We summarize its main components below.

The input is a sequence of words $x_1^J = x_1, \ldots, x_J$, each of them belonging to the source vocabulary V_x and following a one-hot codification scheme. Each word is linearly projected to a fixed-size real-valued vector through an embedding matrix.

The sequence of word embeddings feeds a bidirectional RNN with gated recurrent units (GRU). Therefore, the sequence of word embeddings is analysed in both time directions. This bidirectional GRU network computes a sequence of hidden states $\mathbf{h}_1^J = \mathbf{h}_1, \ldots, \mathbf{h}_J$.

The model features an attention mechanism (Bahdanau et al., 2015), which allows the decoder to selectively focus on parts of the input sequence. Complex relationships in long sentences are therefore kept. At each decoding time-step i, the attention mechanism computes a different context vector $\mathbf{c}_i$ as the weighted sum of the sequence of hidden states $\mathbf{h}_1^J$ from the encoder:

$$\mathbf{c}_i = \sum_{j=1}^{J} \alpha_{ij} \mathbf{h}_j \tag{7}$$

where α_{ij} is a weight assigned to each $\mathbf{h}_j$. This weight is computed by means of a soft alignment model, which measures how well the inputs from the source position j and the outputs at target position i match. Fig. 3 shows the architecture of an attention-based NMT system.

The decoder is another GRU network, that generates the translated sentence $y_1^I = y_1, \ldots, y_I$. The hidden state of the decoder ($\mathbf{s}_i$) depends on the previously generated word y_{i-1}, its previous hidden state ($\mathbf{s}_{i-1}$) and the context vector $\mathbf{c}_i$ from the attention model. Assuming a model for $\Pr(y_i \mid y_1^{i-1}, x_1^J)$ in Eq. (6) of parameters θ (the different matrices in the encoder and the decoder), the probability of a word at the times-step i is defined as:

$$p(y_i \mid y_1^{i-1}, x_1^J; \theta) \;=\; \bar{\mathbf{y}}_i^{\top} \boldsymbol{\varphi}(\mathbf{V}\; \boldsymbol{\phi}(y_{i-1}, \mathbf{s}_i, \mathbf{c}_i)) \tag{8}$$

where $\boldsymbol{\varphi}(\cdot) \in \mathbb{R}^{|V_y|}$ is a softmax function that produces a vector of probabilities, $|V_y|$ is the size of the target

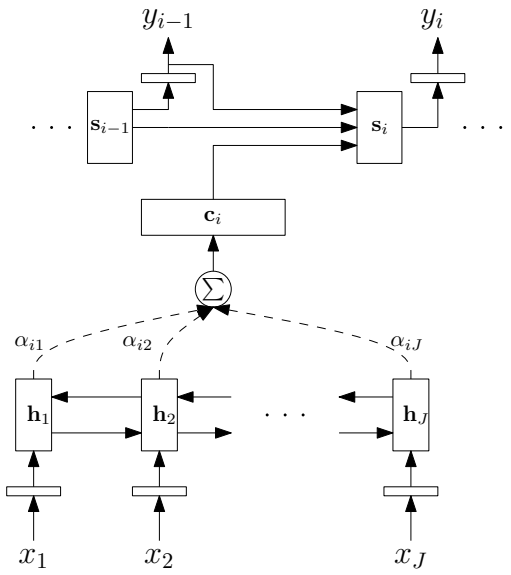

Figure 3: Architecture of the neural machine translation system, equipped with an attentional model, similarly depicted as Bahdanau et al. (2015). The bidirectional RNN encoder processes the source sentence x_1^J and generates the sequence of hidden states $\mathbf{h}_1^J$. The alignment model assigns a weight α_{ij} to each hidden state, according to the previous state of the decoder $\mathbf{s}_{i-1}$ and the hidden state from the encoder. The states from the encoder are then combined in order to obtain a context vector $\mathbf{c}_i$. The decoder generates the next word y_i according to the context vector $\mathbf{c}_i$, its previous hidden state $\mathbf{s}_{i-1}$ and the previously generated word y_{i-1}.

vocabulary, $\bar{\mathbf{y}}_i \in \mathbb{N}^{|V_y|}$ is the one-hot representation of the word y_i, $\mathbf{V} \in \mathbb{R}^{|V_y| \times L}$ is the weight matrix and $\boldsymbol{\phi}$ is the output of an Elman RNN (Elman, 1990) with GRU units and an L-sized maxout (Goodfellow et al., 2013) output layer.

3.1. Training and decoding

To estimate the model parameters θ, the training objective is to maximize the log-likelihood over a bilingual parallel corpus $\mathcal{S} = \{(x^{(s)}, y^{(s)})\}_{s=1}^{S}$, consisting of S sentence pairs, with respect to θ:

$$\hat{\theta} = \arg\max_{\theta} \sum_{s=1}^{S} \sum_{i=1}^{I_s} \log(p(y_i^{(s)} \mid y_1^{i-1(s)}, x^{(s)}; \theta)) \tag{9}$$

where I_s is the length of the s-th target sentence. For the sake of clarity, we drop the dependencies on θ in the rest of equations of this manuscript.

Once the model parameters are estimated, the goal of the NMT system is the same as classical SMT (Eq. (1)). An optimal solution would require to search over the space of all possible target sentence, which is unaffordable. For generating translations, suboptimal decoding strategies such as beam-search were used by Bahdanau et al. (2015), Cho et al. (2014) and Sutskever et al. (2014).

4. Interactive neural machine translation

The addition of interactive mechanisms to NMT systems affects the search process: the search space must be constrained, in order to take into account the user feedback and generate compatible hypotheses.

In this section, we describe the modifications of the search process that interactive scenarios require. Following the statements from Section 2, we distinguish between prefix-based and segment-based interactive protocols.

4.1. Prefix-based interactivity

In this protocol, the user corrects the left-most wrong word of the system hypothesis. Given a translation hypothesis $y_1^I = y_1, \ldots, y_I$, the feedback given to the system has the form $f = \hat{y}_1^i$, where $\hat{y}_1^i$ is the validated prefix together with the corrected word. The inclusion of this feedback into the NMT systems is natural, because sentences are generated from the left to the right. Given a prefix $\hat{y}_1^i$, only a single path accounts for it. The branching of the search process starts once this path has been covered. Introducing the user feedback $f = \hat{y}_1^i$, Eq. (8) becomes:

$$p(y_{i'} \mid y_1^{i'-1}, x_1^J, f = \hat{y}_1^i) = \begin{cases} \delta(y_{i'}, \hat{y}_{i'}), & \text{if } i' \leq i \\ \bar{\mathbf{y}}_{i'}^\top \boldsymbol{\varphi}(\mathbf{V}\ \boldsymbol{\phi}(y_{i'-1}, \mathbf{s}_{i'}, \mathbf{c}_{i'})) & \text{otherwise} \end{cases} \tag{10}$$

where $\delta(\cdot, \cdot)$ is the Kronecker delta. This can be seen as generating the most probable suffix given a validated prefix, and fits into the statistical framework deployed by Barrachina et al. (2009).

4.2. Segment-based interactivity

In the segment-based interactive protocol, the user can perform two actions: introduce a word correction and validate segments to keep in future iterations. A validated segment f is defined over the translation hypothesis y_1^I as a sequence of one or more consecutive target language words: $f = y_i^{i'}$ being $1 \leq i \leq i' \leq I$.

The feedback signal has now the form $f_1^N = f_1, \ldots, f_N$, where $f_1, \ldots, f_N$ is a sequence of N non-overlapping segments validated by the user. The word correction introduced by the user is inserted as a one-word segment in f_1^N. Note that the prefix-based approach is a particular case of the segment-based one, in which the sequence of validated segments only contains the validated prefix (including the corrected word).

Once again, the system must generate a new hypothesis compatible with the feedback signal. For achieving this, the problem is reformulated as the generation of the optimal sequence of non-validated segments $\tilde{g}_1^N = \tilde{g}_1, \ldots, \tilde{g}_N$, where each $\tilde{g}_n$ is a subsequence of words in the target language. The goal is to obtain a sequence of non-validated segments such that its combination with the sequence of validated segments provide a (hopefully) better translation y' according to Eq. (5).

Unlike the prefix-based approach, positions of validated segments f_n in the next hypothesis are unknown beforehand: the user only validates segments of words, not positions in the hypothesis. Therefore, validated segments cannot be introduced on a rigid way as in Eq. (10). The search process must be constrained on a softer way.

We propose to allow the model to decide whether the search process should be constrained or unconstrained. f_1^N and $\tilde{g}_1^N$ are non-overlapping sequences. Hence, the words produced by the system exclusively belong either to a validated or to a non-validated segment. We can differentiate the word generation process according to whether we are generating words belonging to a validated segment or to a non-validated one.

In the first case, the word generation is similar to the prefix-based approach. Let f_n be the n-th validated segment and let i_n be the previous position in y where f_n should start. The word probability expression for the words belonging to this validated segment is:

$$p(y_{i_n+i'} \mid y_1^{i_n+i'-1}, x_1^J, f_1^N) = \delta(y_{i_n+i'}, f_{ni'}), \quad 1 \leq i' \leq |f_n| \tag{11}$$

where $|f_n|$ is the length of the validated segment f_n and $f_{ni'}$ refers to the i'-th word of such segment.

In the second case, words belong to a non-validated segment. This is harder for the NMT system, because the length of the current $\tilde{g}_n$ is unknown. This length needs to be estimated. Let W_n be the length of a non-validated segment g_n, located between two validated segments, f_n and f_{n+1}. Each alternative hypothesis y will (partially) have the form $y = \ldots, f_n, g_n, f_{n+1}, \ldots$. For estimating the value of W_n, we "look forward" and generate $M + 1$ alternatives, each one having a different value of W_n ($0 \leq W_n \leq M$). M is defined as the maximum length of a non-validated segment and it is estimated on a development set. We take the value of W_n that provides the most probable hypothesis, normalised by the length of the non-validated segment.

As before, let i_n be the previous position in y where g_n should start, i.e. the end position of f_n. The value of W_n is estimated as follows:

$$\widehat{W}_n = \underset{0 \leq W_n \leq M}{\arg\max} \frac{1}{W_n + 1} \prod_{i'=i_n+1}^{i_n+W_n+1} p(y_{i'} \mid y_1^{i'-1}, x_1^J) \tag{12}$$

Note that this expression allows the model to generate non-validated segments of length zero, resulting in the consecutive positioning of the validated segments f_n and f_{n+1}.

Once the length of the non-validated segment has been estimated, words belonging to that segment are generated following the regular procedure:

$$p(y_{i_n+i'} \mid y_1^{i_n+i'-1}, x_1^J, f_1^N) = \mathbf{y}_{i_n+i'}^\top \; \boldsymbol{\varphi}(\mathbf{V} \; \boldsymbol{\phi}(y_{i_n+i'-1}, \mathbf{s}_{i_n+i'}, \mathbf{c}_{i_n+i'})), \quad 1 \leq i' \leq \widehat{W}_n \tag{13}$$

We can reuse the computations made in Eq. (12) for evaluating this last expression without additional cost. This gives the model freedom to choose the number of words to insert between validated segments. The system is able to shift forward or backward the validated segments along the new hypothesis. In order to avoid the repetition of words, if the system started to generate words in a non-validated segment $\tilde{g}_n$ that belong to the next validated segment f_{n+1}, we include f_{n+1} into $\tilde{g_n}$, collapsing both segments.

When all validated segments have been already generated, the hypothesis is completed following the non-interactive process (Eq. (8)).

5. Experimental framework

In this section we describe the experimental methodology, corpora, evaluation metrics and MT systems used in this work for assessing our proposals.

5.1. Corpora

Following prior IMT works (Barrachina et al., 2009; González-Rubio et al., 2013; Tomás and Casacuberta, 2006), we tested the proposed methods in five different translation tasks. *Xerox* (Barrachina et al., 2009) consists in user manuals from *Xerox* printers. *EU* (Barrachina et al., 2009) is a collection of proceedings extracted from the Bulletin of the European Union. *TED* (Federico et al., 2011) is a collection of TED talks. *EMEA* (Tiedemann, 2009) is a medical corpus extracted from the *European MEdical Agency*. We used the same data splits that the aforementioned IMT works; hence, the results obtained are directly comparable. Finally, we also used the release v7 of the *Europarl* (Koehn, 2005) corpus, a large collection of proceedings from the European Parliament. We used the `news-test2012` and `news-discuss-test2015` sets as development and test partitions, respectively.

We tokenised all corpora using the standard tool from the `Moses` toolkit (Koehn et al., 2007). We kept sentences truecased, except for the Zh–En language pair, since Chinese has no case information. Table 1 shows the languages and main features of each corpus.

5.2. User simulation

A direct evaluation of an IMT system would require to conduct experiments with human users. Unfortunately, this is excessively costly and slow for deploying IMT prototypes. Hence, automatic evaluation methodologies have been developed in the literature (Barrachina et al., 2009; González-Rubio et al., 2013; Tomás and Casacuberta, 2006), aiming to simulate the behaviour of a real user. We followed such protocols for evaluating our proposal. Nevertheless, it is planned to perform a human evaluation of the system as a future work.

In the simulation, it is assumed that the reference sentences from our parallel corpora are the outputs desired by the user. This is a pessimistic assumption, since the user can find appropriate different translations for the same sentence. We consider only one translation to be the correct one.

Table 1: Main figures of the Xerox, EU, EMEA, Europarl and TED corpora. **S**, **T** and **V** account for number of sentences, number of tokens and vocabulary size respectively. k and M stand for thousands and millions.

		Training			**Development**			**Test**		
		S	**T**	**V**	**S**	**T**	**V**	**S**	**T**	**V**
Xerox	Es	56k	747k	17k	1,025	16k	2k	1,125	10k	2k
	En		662k	15k		14k	2k		8k	2k
EU	En	214k	5.9M	97k	400	12k	3k	800	23k	5k
	En		5.2M	84k		10k	3k		20k	4k
EMEA	Fr	1.1M	17.0M	79k	500	12k	3k	1,000	26k	5k
	En		14.3M	70k		10k	3k		21k	4k
Europarl	Fr	2.0M	61.3M	153k	3,003	82k	12k	1,500	30k	6k
	En		55.7M	134k		73k	20k		27k	6k
TED	Zh	107k	1.9M	42k	934	42k	4k	1,664	33k	4k
	En		2.0k	52k		52k	3k		32k	4k

We defined two simulated scenarios, accounting for both interactive protocols: prefix-based and segment-based. In the first one, the simulated user corrects the first wrong word from the translation hypothesis. The system then, produces an alternative hypothesis, containing the validated prefix. This process continues until the hypothesis produced by the system is the one desired by the user i.e., our reference sentence.

In the segment-based interactivity, the simulated user selects the correct segments from the hypothesis. We assume, without loss of generality, that word corrections are made from the left to the right. The user also inserts word corrections. The system generates an alternative translation hypothesis when a word is inserted. Such hypothesis accounts both for the word correction and the validated segments from previous interactions.

5.3. Assessment metrics

Automatic evaluation of MT systems is still an open problem, and even more in the interactive cases, as many factors affect the translation process. Since a human is involved all along the process, we should assess both the quality of the MT system and the effort required by the user for post-editing the system outputs. The goal of IMT is to minimize the effort made by the user, rather than improving the quality of the system outputs, because final post-edited translations are supposed to be perfect. Hence, we divide the evaluation metrics depending on what they measure: *translation quality* or *human effort*.

5.3.1. Measuring translation quality

The quality of an MT engine is usually estimated by comparing the output of the system with the reference corpus. For evaluating our MT systems, we used the following well-known metrics:

BiLingual Evaluation Understudy (BLEU) (Papineni et al., 2002): is the most widely used metric for evaluating MT systems. BLEU aims to model the correspondence between the translation generated by the MT system and the one produced by a human translator. It is defined as the geometric mean of the n-gram precision between the hypothesis and the reference sentences, modified by a brevity penalty.

Translation Edit Rate (TER) (Snover et al., 2006): is the minimum number of word edit operations (insertion, substitution, deletion and swapping between groups of words) that must be made in order to transform the hypothesis into the reference. The number of edit operations is normalised by the number of words in the reference sentence. TER can be used as a representation of the human-targeted TER (HTER) when considering the reference sentences as the human post-edited version of the MT translation (Zaidan and Callison-Burch, 2010).

5.3.2. Measuring human effort

The human effort required for post-editing the output of an MT system is usually estimated as the number of actions performed by the human agent in order to obtain the desired translations. Our protocol supports two different input devices: keyboard and mouse. We implemented two metrics that measure the use of each device:

Word Stroke Ratio (WSR) (Tomás and Casacuberta, 2006): number of word corrections made by the user for obtaining the desired translation from the system hypothesis, divided by the total number of words in the final sentence. It is assumed that the cost of correcting a word is constant, regardless its length.

Mouse Action Ratio (MAR) (Barrachina et al., 2009): measures the effort made by the user with the mouse during the interactive PE process. It is defined as the number of mouse actions made by the user, divided by the total number of characters in the final hypothesis. We count mouse "clicks" as mouse actions. Correcting a wrong word requires to select the word and accounts for one mouse action. Validating a segment requires two mouse actions: clicking at the beginning and at the end of the segment. A single mouse action is enough for validating one-word segments. We add an additional action per sentence, accounting for the final validation of a hypothesis.

We take WSR and MAR as a broad approximation of the human effort required by using the IMT system. Nevertheless, for properly measuring the user effort, a human experimentation should be conducted.

5.4. MT systems

We used the `GroundHog` toolkit[1] for training the NMT systems. `GroundHog` is built upon `Theano` (Theano Development Team, 2016). The search process was implemented in the Python scripting language.

We followed the architecture introduced by Bahdanau et al. (2015) and described in Section 3. The main model hyperparameters for each task were obtained through the random search (Bergstra and Bengio, 2012) method. Such parameters were the size of the word embedding (tested in the range $[100, 620]$) and the number of GRU hidden units (tested in the range $[100, 1000]$). We set the same configuration for both encoder and decoder networks at each run. We chose the model with highest BLEU on the development set. Figures of these hyperparameters are detailed in Table 2. Since a random seach for the Europarl task was excessively costly, we followed Bahdanau et al. (2015) and used a word embedding size of 620 and 1000 GRU units for each RNN.

Table 2: Main NMT hyperparameters figures, obtained through the random search method: size of the word embedding and size of the GRU hidden state. The same word embedding and GRU sizes have been used in both encoder and decoder RNNs, for each configuration.

		Word embedding	GRU units
Xerox	Es–En	289	354
	En–Es	415	607
EU	Es–En	456	684
	En–Es	382	712
EMEA	Fr–En	511	735
	En–Fr	385	452
TED	Zh–En	477	560
	En–Zh	538	611

The rest of specific hyperparameters were set according to preliminary trials. Source and target vocabularies sizes were set to 30,000 words, except for the Xerox task, where we could process the full vocabulary.

[1] https://github.com/lisa-groundhog/GroundHog

For dealing with unknown words, we used the strategy described by Jean et al. (2015), consisting in replacing the unknown word by its correspondingly aligned source word. Models were trained on an *Nvidia Titan X* GPU. We used minibatch SGD for updating the parameters, with minibatches of 80 sentences. The Adadelta (Zeiler, 2012) method automatically set the learning rate. We computed the BLEU on the development set each 2,000 updates. We early-stopped the training when this BLEU did not improve for 20,000 updates. The beam size was fixed to 6.

We compare the NMT technology against the classical PB approach. As PB system, we used the standard configuration of `Moses`. This includes bilingual phrase, language, distortion and reordering models, among others. We used a 5-gram language model smoothed with the Kneser-Ney method (Kneser and Ney, 1995) and estimated with the `SRILM` toolkit (Stolcke, 2002). We estimated the weights of the log-linear model using the *Minimum Error Rate Training* (MERT) (Och, 2003) method on the development partition.

PB models were used to obtain a baseline for prefix-based IMT. We followed the procedure described by Barrachina et al. (2009) for exploring a word graph and generating the best suffix for a given prefix. For each sentence to translate, we generated a word graph with the PB system. The word graph was treated as a weighted finite-state automaton. We parsed the prefix over the word graph to find the best path that accounts for the prefix, going from the initial state to any other intermediate state. Finally, we obtained the corresponding translation for the best path from this intermediate state to the final state. Therefore, our implementation of prefix-based IMT is consistent with Barrachina et al. (2009), but considering that we generated word graphs with the current SMT state-of-the-art `Moses` toolkit.

6. Results and discussion

This section presents the results of the experiments conducted for assessing our proposals. In this section we only show the results obtained on the test set. First, we compare the neural system against the PB system in a classical PE scenario. Next, we assess the performance of both systems in the prefix-based IMT. Finally, we compare the proposed segment-based IMT approach, showing that the typing effort can be significantly reduced. For all metrics, we report their confidence intervals ($p = 0.05$), computed by means of bootstrap resampling (Koehn, 2004).

6.1. Quantitative analysis

Table 3 shows the results of translation quality of the neural and phrase-based systems. Using BLEU to measure the translation quality, the neural approach generally performs slightly worse than PB systems. We conjecture that part of these differences are due to the vocabulary reduction and rare words handling. The inclusion of larger vocabularies (Jean et al., 2015) or the use of subword units (Sennrich et al., 2016) may bring closer the performance of both approaches. TER presents a similar behaviour: PB technology also performs slightly better than the neural one in some tasks, although differences in TER are generally smaller than those observed in BLEU.

As we move on to the interactive scenario, NMT exhibits its strengths. Table 4 shows the results of the prefix-based interaction. Despite having a poorer general performance in pure automatic translation, the neural approach significantly outperforms PB systems in all cases. Differences yield from almost 5 WSR points up to 15, which accounts for reductions of the 30% of the typing effort. Similar decreases are obtained in the MAR values.

These WSR and MAR differences are probably due to the different nature of both approaches: neural models are naturally smoother than the PB ones, and they react better to the introduction of word corrections. The modifications over the search process required by the interactive protocol are small, hence, they handle on a more natural way changes in their hypotheses. On the other hand, PB systems are more rigid, relying on the search over a pruned word graph.

Moreover, the insertion of an unexpected correction (e.g. an unknown word) can produce a failure of the PB system. If the user feedback leads to a specific state in the word graph that has no path to any final state, the system fails. An error correction method is used for overcoming this issue (Barrachina et al., 2009), but it may lead to successive wrong hypotheses. NMT systems handle the out-of-vocabulary problem by

Table 3: Results of translation quality for all tasks in terms of BLEU [%] and TER [%]. Best results for each task and metric are bold-faced.

		BLEU [%]		**TER** [%]	
		PB	**NMT**	**PB**	**NMT**
Xerox	Es–En	51.3 ± 2.3	$\mathbf{53.2 \pm 2.5}$	32.5 ± 1.8	$\mathbf{30.0 \pm 1.7}$
	En–Es	$\mathbf{60.0 \pm 2.3}$	58.5 ± 2.4	$\mathbf{29.1 \pm 1.7}$	29.2 ± 1.7
EU	Es–En	$\mathbf{46.4 \pm 1.9}$	40.7 ± 1.8	$\mathbf{41.3 \pm 1.6}$	45.2 ± 1.6
	En–Es	$\mathbf{45.8 \pm 1.8}$	41.4 ± 1.7	$\mathbf{43.3 \pm 1.6}$	46.2 ± 1.5
EMEA	Fr–En	$\mathbf{30.1 \pm 1.1}$	26.3 ± 1.1	$\mathbf{48.9 \pm 1.1}$	53.7 ± 1.2
	En–Fr	$\mathbf{29.7 \pm 1.0}$	26.1 ± 1.3	$\mathbf{52.6 \pm 1.0}$	53.5 ± 1.7
Europarl	Fr–En	$\mathbf{23.0 \pm 0.9}$	20.8 ± 1.0	59.9 ± 1.0	60.7 ± 1.2
	En–Fr	$\mathbf{22.7 \pm 0.9}$	21.0 ± 0.9	$\mathbf{59.5 \pm 1.0}$	61.1 ± 1.2
TED	Zh–En	$\mathbf{11.6 \pm 0.6}$	10.7 ± 0.6	$\mathbf{76.0 \pm 1.4}$	77.4 ± 1.1
	En–Zh	$\mathbf{8.7 \pm 0.5}$	7.8 ± 0.5	82.7 ± 1.0	$\mathbf{81.9 \pm 1.0}$

Table 4: Results of prefix-based interaction for all tasks, measured in WSR [%] and MAR[%]. Best results for each task and metric are bold-faced.

		WSR [%]		**MAR** [%]	
		PB	**NMT**	**PB**	**NMT**
Xerox	Es–En	29.7 ± 1.8	$\mathbf{24.0 \pm 1.3}$	10.6 ± 0.6	$\mathbf{9.8 \pm 0.4}$
	En–Es	31.1 ± 1.5	$\mathbf{20.9 \pm 1.1}$	9.6 ± 0.4	$\mathbf{7.9 \pm 0.3}$
EU	Es–En	45.6 ± 1.8	$\mathbf{35.1 \pm 1.4}$	10.5 ± 0.4	$\mathbf{8.2 \pm 0.3}$
	En–Es	45.7 ± 1.7	$\mathbf{34.6 \pm 1.3}$	10.3 ± 0.4	$\mathbf{7.9 \pm 0.3}$
EMEA	Fr–En	58.8 ± 1.3	$\mathbf{49.7 \pm 1.1}$	12.9 ± 0.3	$\mathbf{11.0 \pm 0.2}$
	En–Fr	58.7 ± 1.3	$\mathbf{43.1 \pm 0.9}$	13.5 ± 0.3	$\mathbf{10.4 \pm 0.2}$
Europarl	Fr–En	70.3 ± 1.3	$\mathbf{52.6 \pm 1.0}$	19.5 ± 0.4	$\mathbf{15.5 \pm 0.3}$
	En–Fr	67.7 ± 1.2	$\mathbf{56.6 \pm 1.0}$	17.8 ± 0.3	$\mathbf{15.4 \pm 0.3}$
TED	Zh–En	81.3 ± 1.0	$\mathbf{61.7 \pm 0.9}$	23.3 ± 0.3	$\mathbf{18.5 \pm 0.3}$
	En–Zh	83.4 ± 1.5	$\mathbf{71.0 \pm 0.8}$	59.2 ± 1.0	$\mathbf{49.7 \pm 0.5}$

mapping unknown words to a special token. Since during the training phase the system has seen unknown words, it is prepared to deal with this inconvenience.

Table 5 shows the segment-based interaction results, compared with the prefix-based neural IMT. Using this protocol, the typing effort is always diminished, at the expense of more mouse interactions. The WSR is reduced by a factor ranging from 7% up to 15%. On the other hand, MAR values are increased in all tasks.

This is an expected behaviour: since the user validates segments, the mouse activity is higher. However, using the mouse for validating segments avoids to introduce errors in successive iterations on correct parts of the hypothesis. Therefore, this mouse activity increase may pay off from the user comfortability point of view. Nevertheless, due to the flexibility of the segment-based approach, users are free to choose how to use the system: a user may prefer to use more intensively the mouse, while another one may prefer to introduce more word corrections. Both user behaviours are supported by the protocol.

Despite that the segment-based approach obtains better WSR than the prefix-based protocol for every task, the differences between them are lower than those achieved by the neural model with respect to the PB approach. This is probably due to the way in which the neural system generates the translations: since

Table 5: Results of neural segment-based interaction for all tasks, compared with the neural prefix-based approach. Best results for each task and metric are bold-faced.

		WSR [%]		**MAR** [%]	
		Prefix	**Segment-based**	**Prefix**	**Segment-based**
Xerox	Es–En	24.0 ± 1.3	$\mathbf{21.8 \pm 1.2}$	$\mathbf{9.8 \pm 0.4}$	11.9 ± 0.6
	En–Es	20.9 ± 1.1	$\mathbf{19.4 \pm 1.0}$	$\mathbf{7.9 \pm 0.3}$	13.2 ± 0.5
EU	Es–En	35.1 ± 1.4	$\mathbf{30.5 \pm 1.3}$	$\mathbf{8.2 \pm 0.3}$	13.7 ± 0.4
	En–Es	34.6 ± 1.3	$\mathbf{30.3 \pm 1.2}$	$\mathbf{7.9 \pm 0.3}$	12.7 ± 0.4
EMEA	Fr–En	49.7 ± 1.1	$\mathbf{42.6 \pm 1.0}$	$\mathbf{11.0 \pm 0.2}$	17.5 ± 0.3
	En–Fr	43.1 ± 0.9	$\mathbf{38.5 \pm 0.8}$	$\mathbf{10.4 \pm 0.2}$	16.7 ± 0.3
Europarl	Fr–En	52.6 ± 1.0	$\mathbf{48.3 \pm 0.9}$	$\mathbf{15.5 \pm 0.3}$	23.8 ± 0.4
	En–Fr	56.6 ± 1.0	$\mathbf{51.9 \pm 0.9}$	$\mathbf{15.4 \pm 0.3}$	22.2 ± 0.3
TED	Zh–En	61.7 ± 0.9	$\mathbf{55.0 \pm 0.8}$	$\mathbf{18.5 \pm 0.3}$	25.6 ± 0.3
	En–Zh	71.0 ± 0.8	$\mathbf{61.7 \pm 0.8}$	$\mathbf{49.7 \pm 0.5}$	66.8 ± 0.7

it follows a left-to-right direction, the prefix-based approach fits nicely into it. The inclusion of the segment-based feedback is more unnatural. Although it helps the system, in light of the results, the benefits obtained are limited.

However, the segment-based framework allows the implementation of more complex user models. For instance, the user could remove words between validated segments, or drop all non-validated segments from a hypothesis, achieving the desired translation in fewer interactions. We think that higher gains could be obtained by sophisticating the user model.

Finally, the average response times for all systems are shown at Table 6. In the neural systems, the response rates are adequate for an interactive scenario (Nielsen, 1993), when using the GPU. If computations are made on the CPU, the response time is increased approximately 10 times. We are aware that, in an industrial setting, the device used will be the CPU, hence, strategies for scaling up the system and enhancing the CPU response time should be developed in the future.

Table 6: Average system response time (in seconds) per interaction, for each task. **NMT-CPU** refers to the execution of the neural system in a CPU, while the column **NMT-GPU** show the times for the GPU execution. In the case of the PB systems, we report the time of the sole exploration of the word graph (**PB-SEARCH**), together with the time required for both building and exploring the word graph (**PB-ALL**). All PB systems run on CPU. Best results for each task are bold-faced.

		NMT-GPU	**NMT-CPU**	**PB-ALL**	**PB-SEARCH**
Xerox	Es–En	**0.04**	0.34	1.16	0.14
	En–Es	**0.09**	0.93	1.09	0.34
EU	Es–En	**0.32**	4.01	1.98	0.73
	En–Es	**0.37**	4.21	2.23	0.98
EMEA	Fr–En	**0.38**	4.03	1.50	0.49
	En–Fr	**0.25**	2.45	0.97	0.36
Europarl	Fr–En	1.11	4.22	0.91	**0.26**
	En–Fr	1.13	4.06	1.25	**0.38**
TED	Zh–En	0.28	2.89	0.68	**0.25**
	En–Zh	**0.26**	2.77	0.87	0.33

In the case of the PB systems, we measure two different scenarios, depending on whether the word graph is pre-computed or not. In the first case, response times are acceptable, allowing fluent interactivity. In the second case, response times are inadequate for the interactivity. Nevertheless, note that a comparison

of the NMT system with the first scenario is unfair, because in the NMT system we do not assume any pre-computation. It should be investigated if advancing computations may help to reduce the responsiveness of the NMT system.

Due to the different metrics and pre/post-processing techniques (e.g. corpus categorisation) employed in the literature, it is hard to directly compare the obtained results. For computing the PB results, we use the software kindly provided by the authors of Barrachina et al. (2009). Hence, these results can be considered an updated version of Barrachina et al. (2009), using the current `Moses` implementation.

On the other hand, several works tackled the same tasks. In Tomás and Casacuberta (2006), it was developed a specific decoder for IMT. They obtained a WSR of 30.0% and 28.7% in the Xerox task, for the En–Es and Es–En language directions, respectively. Our prefix-based neural system provided similar results. The segment-based protocol significantly reduces the keyboard activity, at the expense of more mouse activity. González-Rubio et al. (2010b) studied the inclusion of confidence measures in the prefix-based interactive process. Their base IMT system obtained a WSR of 52%, which is clearly outperformed by the neural system. Finally, González-Rubio et al. (2016) pioneered the segment-based idea and developed a system based on hierarchical models and hypergraphs. This was tested on a categorised version of the EU task. They reported a WSR of 54.5% and 35.1% for the prefix-based and segment-based protocols, respectively. The neural system also outperforms these results (35.1% and 30.5%). Moreover, they reported an average response time of 3 seconds. The NMT system response time is approximately 10 times lower when executed on GPU, which is closer to the demands of interactive systems.

6.2. Qualitative analysis: IMT session examples

Now, we qualitatively compare the behaviour of the PB systems against the neural technology with a typical example from the EMEA corpus. We show the different IMT sessions of the PB and neural systems, in order to translate a French sentence into English. The source sentence is "*Ils seront invités à fournir un échantillon d' urine propre .*" and the corresponding target sentence is "*They will be asked to provide a clean catch urine sample .*". Since these are real examples, we present all sentences tokenised.

As shown in Fig. 4, the PB system requires 6 interactions in order to correctly translate the source sentence. Note that, at the first iteration, the PB system is unable of properly taking into account the correction provided by the user: the generated suffix contains the word "*requested*", although it should be removed from the hypothesis.

Fig. 5 shows the analogue example, but using the neural system. The first hypothesis is similar to that provided by the PB system, and the first correction is the same. But now, the NMT system adapts the next hypothesis to the user feedback: The system processes the correction "*asked*" as a synonym for "*encouraged*". Therefore, it is able to provide a coherent hypothesis. With this, the number required of iterations is reduced from 6 to 4.

Source (x): Ils seront invités à fournir un échantillon d' urine propre .

Target translation ($\hat{y}$): They will be asked to provide a clean catch urine sample .

IT-0	**MT**	They will be invited to provide a panel of urine clean .
IT-1	**User**	*They will be* **asked** to provide a panel of urine clean .
	MT	*They will be asked* requested to provide clean urine sample .
IT-2	**User**	*They will be asked* **to** requested to provide clean urine sample .
	MT	*They will be asked to* provide a panel of urine clean .
IT-3	**User**	*They will be asked to provide a* **clean** panel of urine clean .
	MT	*They will be asked to provide a clean* urine sample .
IT-4	**User**	*They will be asked to provide a clean* **catch** urine sample .
	MT	*They will be asked to provide a clean catch* (urine sample .
IT-5	**User**	*They will be asked to provide a clean catch* **urine** (urine sample .
	MT	*They will be asked to provide a clean catch urine* .
IT-6	**User**	*They will be asked to provide a clean catch urine* **sample** .
	MT	*They will be asked to provide a clean catch urine sample* .
END	**User**	*They will be asked to provide a clean catch urine sample* .

Figure 4: Real PB prefix-based IMT session to translate a sentence from French into English: given the input sentence x, the user desires to obtain the target sentence $\hat{y}$. User corrections are in **bold** font, while validated prefixes are in *italic* font. First, the system generates an initial hypothesis. Following the protocol, at iteration **IT-1**, the user corrects the first wrong word, introducing the word "*asked*" at position 4. Then, the system reacts and suggests a new suffix. In this hypothesis, the words "*asked*" and "*requested*" are consecutive in the new hypothesis. This shows that the PB system is unable to properly deal with the correction given by the user. The next iteration starts then, with the user validating the next wrong word and the system generating a compatible hypothesis. The protocol is repeated until the desired translation is obtained. This IMT process requires 6 word corrections.

Source (x): Ils seront invités à fournir un échantillon d' urine propre .

Target translation ($\hat{y}$): They will be asked to provide a clean catch urine sample .

IT-0	**MT**	They will be encouraged to provide a clean urine sample .
IT-1	**User**	*They will be* **asked** encouraged to provide a clean urine sample .
	MT	*They will be asked* to provide a clean urine sample .
IT-2	**User**	*They will be asked to provide a clean* **catch** urine sample .
	MT	*They will be asked to provide a clean catch* .
IT-3	**User**	*They will be asked to provide a clean catch* **urine** .
	MT	*They will be asked to provide a clean catch urine* .
IT-4	**User**	*They will be asked to provide a clean catch urine* **sample** .
	MT	*They will be asked to provide a clean catch urine sample* .
END	**User**	*They will be asked to provide a clean catch urine sample* .

Figure 5: Real neural prefix-based IMT session, considering the same sentence, protocol, and format as in Fig. 4, but using the NMT system. In this case, the NMT engine is able to cope with the given prefix at the first iteration, obtaining a meaningful hypothesis at this iteration. The protocol is the same as before but, due to the behaviour of the NMT system, the session ends after 4 interactions, which accounts for a reduction of the 33% of the effort, compared with the PB system.

At **IT-1** of Fig. 5, the NMT system produced the correct segment "urine sample .". Nevertheless, this segment is lost at **IT-2**. The segment-based protocol aims to prevent this behaviour. Fig. 6 shows addition of the segment validation. Now, at the first iteration, the user validates three segments and introduces a word correction ("*asked*"). The user feedback allows the model to keep into the hypothesis the validated segment "*urine sample .*", at the second iteration. Since there are no new segments, at the next iteration, the user only introduces a word correction. Since the hypothesis provided by the system is now correct, the

user accepts it as final translation. Using the segment-based approach, only two interactions are necessary, which accounts for a reduction of the 66% and the 50% of the number of interactions from prefix-based PB and NMT systems, respectively.

Source (x): Ils seront invités à fournir un échantillon d' urine propre .

Target translation ($\hat{y}$): They will be asked to provide a clean catch urine sample .

IT-0	**MT**	They will be encouraged to provide a clean urine sample .
IT-1	**User**	They will be **asked** encouraged to provide a clean urine sample .
	MT	They will be asked to provide a clean urine sample .
IT-2	**User**	They will be asked to provide a clean **catch** urine sample .
	MT	They will be asked to provide a clean catch urine sample .
END	**User**	*They will be invited to provide a clean urine sample .*

Figure 6: Real neural segment-based IMT session. Same sentence, system, and format as in Fig. 5, but following the segment-based protocol. Boxed text represents validated segments. Now, in addition to correcting words, the user validates with the mouse correct segments in the hypotheses. Hence, at iteration **IT-1**, the user first selects the correct segments of the hypothesis and then introduces a word correction ("*asked*"), which is included as a one-word validated segment. Next, the system provides a new hypothesis compatible with the sequence of validated segments. At iteration **IT-2**, since there are no new correct segments, the user only introduces the word correction "*and*". The system generates another hypothesis, which is completely correct. Finally, the user validates this hypothesis, finishing the process. Only 2 iterations are required during this session. In comparison, the prefix-based approaches require 6 and 4 iterations in the case of the PB and neural systems, respectively.

For better understanding some of the weaknesses of our approach, we study now some failure cases of the neural IMT system. First, we show a case where the corrections made by the user are somewhat unexpected by the system. The source sentence is "*Paquete de 6 (contiene 6 cartuchos)*" and the target sentence is "*6 Pack (contains 6 cartridges)*". The initial hypothesis of the NMT system is "*Zone 6 (assume 6 cartridges)*". A traditional post-editing strategy would require three corrections. Fig. 7 shows the post-editing session of the neural prefix-based IMT system. As the successive corrections are inserted, the system introduces some artefacts in the hypotheses, which demand further corrections.

Source (x): Paquete de 6 (contiene 6 cartuchos)

Target translation ($\hat{y}$): 6 Pack (contains 6 cartridges)

IT-0	**MT**	Zone 6 (assume 6 cartridges)
IT-1	**User**	**6** Zone 6 (assume 6 cartridges)
	MT	*6* Thermistor 6 (three cartridges)
IT-2	**User**	*6* **Pack** Thermistor 6 (three cartridges)
	MT	*6 Pack* 6 (containing cartridges)
IT-3	**User**	*6 Pack* **(** 6 (containing cartridges)
	MT	*6 Pack (* 10 cartridges)
IT-4	**User**	*6 Pack (* **contains** 10 cartridges)
	MT	*6 Pack (contains* 10 cartridges)
IT-5	**User**	*6 Pack (contains* **6** 10 cartridges)
	MT	*6 Pack (contains 6* 9)
IT-6	**User**	*6 Pack (contains 6* **cartridges** 9)
	MT	*6 Pack (contains 6 cartridges*)
END	**User**	*6 Pack (contains 6 cartridges)*

Figure 7: Real failure case of a neural prefix-based IMT session. The user corrections are unexpected for the system, which steadily produces wrong hypotheses.

Note that in the previous example, the original hypothesis (at **IT-0**) contained correct sequences of words which were removed from the next hypotheses. The segment-based approach (Fig. 8) intends to avoid this phenomenon. In this case, the correct segment " *6 cartridges)*" is kept along the session. Hence, the problem is mitigated, requiring less interactions.

Source (x): Paquete de 6 (contiene 6 cartuchos)
Target translation ($\hat{y}$): 6 Pack (contains 6 cartridges)

IT-0	**MT**	Zone 6 (assume 6 cartridges)
IT-1	**User**	**6** Zone 6 (assume 6 cartridges)
	MT	6 Thermistor 6 (6 cartridges)
IT-2	**User**	6 **Pack** Thermistor 6 (6 cartridges)
	MT	6 Pack 6 (6 cartridges)
IT-3	**User**	6 Pack (6 (6 cartridges)
	MT	6 Pack (6 cartridges)
IT-4	**User**	6 Pack (**contains** 6 cartridges)
	MT	6 Pack (contains 6 cartridges)
END	**User**	*6 Pack (contains 6 cartridges)*

Figure 8: Real failure case of a neural segment-based IMT session. Although the behaviour is similar to Fig. 7, in this case the selection of the correct segments in the hypothesis prevents from the degradation suffered by the prefix-based protocol.

Finally, we show a case where the segment-based approach worsens the regular post-editing and the prefix-based strategy. In this case, the source sentence is "*La Utilidad de administración de fuentes queda eliminada del sistema .*" and its corresponding translation is "*The Font Management Utility is removed from your system .*". Fig. 9 shows the prefix-based session for this example. 2 interactions are required in order to produce the correct translation. A regular post-editing process would also require 2 corrections.

Source (x): La Utilidad de administración de fuentes queda eliminada del sistema .
Target translation ($\hat{y}$): The Font Management Utility is removed from your system .

IT-0	**MT**	The Font Management Utility are removed from the system .
IT-1	**User**	*The Font Management Utility* **is** are removed from the system .
	MT	*The Font Management Utility is* removed from the system .
IT-2	**User**	*The Font Management Utility is removed from* **your** the system .
	MT	*The Font Management Utility is removed from your* system .
END	**User**	*The Font Management Utility is removed from your system .*

Figure 9: Prefix-based neural IMT session. Same notation as in previous examples. Two corrections are required for correcting the initial hypothesis. A regular post-editing process would require the same two corrections.

But as we apply the segment-based protocol, the system duplicates parts of the target sentence (Fig. 10): "*the system*" is erroneously inserted at **IT-1**, when this concept is already considered by the segment system . . This requires an additional correction. This error arises because the system predicts the next words according to the previously generated words and the immediately next segment. Hence, the segment system . is not taken into account when generating "*the system*" at **IT-1**. Although this happens infrequently, most of the errors made in the segment-based protocol are due to this. A more advanced decoding strategy, e.g. bidirectional decoding (Liu et al., 2016), may alleviate this problem.

Source (x): La Utilidad de administración de fuentes queda eliminada del sistema .

Target translation ($\hat{y}$): The Font Management Utility is removed from your system .

IT-0	**MT**	The Font Management Utility are removed from the system .
IT-1	**User**	The Font Management Utility **is** are removed from the system .
	MT	The Font Management Utility is the system removed from the system .
IT-2	**User**	The Font Management Utility is **removed** the system removed from the system .
	MT	The Font Management Utility is removed from the system .
IT-3	**User**	The Font Management Utility is removed from **your** the system .
	MT	The Font Management Utility is removed from your system .
END	**User**	*The Font Management Utility is removed from your system .*

Figure 10: Failure case of a segment-based neural IMT session. Same notation as in previous examples. At **IT-1**, the user selects the segment system . . The new suggestion provided by the system is unable of properly taking this segment into account. Hence, "*the system*" is duplicated at the new hypothesis. This leads to an additional correction.

7. Conclusions and future work

In this work, we proposed and deployed a neural approach to the interactive paradigm of MT. Additionally to the classical left-to-right IMT, we presented an alternative interaction protocol. By allowing the selection of correct parts from the ongoing hypothesis, this protocol aims to offer more freedom to the user. We conducted simulated experiments on different language pairs and domains, with satisfactory results: neural models obtained significant improvements in effort reduction over a state-of-the-art PB system. The inclusion of the new protocol lowered even more the typing effort required by the user. We conjecture that these enhancements are due to the flexibility of the neural models and their capability for adapting to the user feedback.

Moreover, the simplicity of the NMT decoders—compared against the complex PB or hierarchical ones—allows the easier development of new interactive protocols. This enables the future implementation of new protocols, that may lead to increase the translation performance.

The next step to take is to conduct a human-based experimentation which confirms the results obtained in this work. We plan to integrate the neural technology into the `CasMaCat` (Alabau et al., 2013) workbench. For doing that, we want to integrate a GPU into the `CasMaCat` server. In addition, NMT decoding speed should also be accelerated, in order to be able to reach a CPU real-time response rate. Stochastic sampling or a distributed implementation of Noisy Parallel Approximate Decoding (Cho, 2016) may achieve real-time response. Moreover, further research on the NMT technology should be made, for addressing the weaknesses of our approach. Byte Pair Encoding (Sennrich et al., 2016) or the large vocabulary trick (Jean et al., 2015) may help to deal with the treatment of unknown words and the vocabulary limitation. As discussed in Section 6.2, bidirectional target decoding (Liu et al., 2016) may also enhance the predictions of the system, specially in the segment-based scenario.

Other interesting way that opens from this work is the inclusion of multimodal feedback signals. Previous works have already integrated speech (Alabau et al., 2011) or e-pens (Alabau et al., 2014) into the interactive pipeline. Neural networks are able to incorporate signals from different sources. Thus, the inclusion of multimodality could lead to even larger productivity increases. Furthermore, the encoder-decoder framework is not limited only to MT. Hence, it can be applied to other structured prediction tasks, such as handwritten transcription. In the future, the neural interactive framework deployed in this work should be tested in these tasks, hoping to obtain improvements, as in MT.

Nowadays, reinforcement learning (RL) is attracting again the attention of many researchers. Some recent works applied RL techniques to sequence-to-sequence problems (Ranzato et al., 2015; Shen et al., 2015; Xu et al., 2015) and more specifically, to NMT (Wu et al., 2016). Within the interactive scope, it seems clear the existence of a relationship between RL and interactivity. We are unaware of any attempt aiming to bring both worlds together. Nevertheless, we plan to explore the use of RL techniques in the interactive framework in the future.

Finally, both neural networks and IMT fit nicely into the online learning (OL) framework. A validated hypothesis is a new training sample, which can be used by the system for improving its performance. The effectiveness of the combination of the online and interactive paradigms has already been shown in the MT field (Ortiz-Martínez, 2016). In this scenario, OL entails an enhanced user experience, since the system adapts its behaviour to the user preferences. Related to this, active learning (AL) techniques for classical IMT have also been studied (González-Rubio et al., 2012). In a real environment, with massive data streams, the use of AL becomes mandatory. The deployment of online training techniques for interactive NMT would also allow the inclusion of AL into the neural interactive pipeline. In a near future, we plan to apply both online and active learning methods into the interactive NMT approach, hoping to develop a more personal user experience that implies a boost in the industrial translation performance.

Acknowledgements

The authors wish to thank the anonymous reviewers for their careful reading and in-depth criticisms and suggestions. This work was partially founded by the project ALMAMATER (PrometeoII/2014/030). We also acknowledge NVIDIA for the donation of the GPU used in this work.

References

Alabau, V., Bonk, R., Buck, C., Carl, M., Casacuberta, F., García-Martínez, M., González-Rubio, J., Koehn, P., Leiva, L. A., Mesa-Lao, B., Ortiz-Martínez, D., Saint-Amand, H., Sanchis-Trilles, G., Tsoukala, C., 2013. CASMACAT: An open source workbench for advanced computer aided translation. The Prague Bull. of Math. Linguist. 100, 101–112.

Alabau, V., Rodríguez-Ruiz, L., Sanchis, A., Martínez-Gómez, P., Casacuberta, F., 2011. On multimodal interactive machine translation using speech recognition. In: Proceedings of the International Conference on Multimodal Interaction. pp. 129–136.

Alabau, V., Sanchis, A., Casacuberta, F., 2014. Improving on-line handwritten recognition in interactive machine translation. Pattern Recognit. 47 (3), 1217–1228.

Azadi, F., Khadivi, S., 2015. Improved search strategy for interactive machine translation in computer-asisted translation. In: Proceedings of Machine Translation Summit XV. pp. 319–332.

Bahdanau, D., Cho, K., Bengio, Y., 2015. Neural machine translation by jointly learning to align and translate. In: Proceedings of the International Conference on Learning Representations. `arXiv:1409.0473 [cs.CL]`.

Barrachina, S., Bender, O., Casacuberta, F., Civera, J., Cubel, E., Khadivi, S., Lagarda, A., Ney, H., Tomás, J., Vidal, E., Vilar, J.-M., 2009. Statistical approaches to computer-assisted translation. Comput. Linguist. 35, 3–28.

Bender, O., Hasan, S., Vilar, D., Zens, R., Ney, H., 2005. Comparison of generation strategies for interactive machine translation. In: Proceedings of the Annual Conference of the European Association for Machine Translation. pp. 33–40.

Bengio, Y., Ducharme, R., Vincent, P., Janvin, C., 2003. A neural probabilistic language model. Mach. Learn. Res., 1137–1155.

Bergstra, J., Bengio, Y., 2012. Random search for hyper-parameter optimization. The J. of Mach. Learn. Res. 13 (1), 281–305.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Jimeno Yepes, A., Koehn, P., Logacheva, V., Monz, C., Negri, M., Neveol, A., Neves, M., Popel, M., Post, M., Rubino, R., Scarton, C., Specia, L., Turchi, M., Verspoor, K., Zampieri, M., 2016. Findings of the 2016 conference on machine translation. In: Proceedings of the First Conference on Machine Translation. pp. 131–198.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., Mercer, R. L., 1993. The mathematics of statistical machine translation: Parameter estimation. Comput. Linguist. 19 (2), 263–311.

Cai, D., Zhang, H., Ye, N., 2013. Improvements in statistical phrase-based interactive machine translation. In: Proceedings of the International Conference on Asian Language Processing. pp. 91–94.

Castaño, M.-A., Casacuberta, F., 1997. A connectionist approach to machine translation. In: Proceedings of the International Conference on Theoretical and Methodological Issues in Machine Translation. pp. 160–167.

Cho, K., 2016. Noisy parallel approximate decoding for conditional recurrent language model, `arXiv:1605.03835 [cs.CL]`.

Cho, K., van Merriënboer, B., Bahdanau, D., Bengio, Y., 2014. On the properties of neural machine translation: Encoder-decoder approaches. In: Proceedings of the Workshop on Syntax, Semantic and Structure in Statistical Translation. pp. 103–111.

Chung, J., Cho, K., Bengio, Y., 2016. A character-level decoder without explicit segmentation for neural machine translation, `arXiv:1603.06147 [cs.CL]`.

Costa-Jussà, M. R., Fonollosa, J. A. R., 2016. Character-based neural machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 357–361.

Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., Makhoul, J., 2014. Fast and robust neural network joint models for statistical machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 1370–1380.

Elman, J. L., 1990. Finding structure in time. Cogn. Sci. 14 (2), 179–211.

Federico, M., Bentivogli, L., Paul, M., Stüker, S., 2011. Overview of the IWSLT evaluation campaign. In: Proceedings of the International Workshop on Spoken Language Translation. pp. 11–27.

Foster, G., Isabelle, P., Plamondon, P., 1997. Target-text mediated interactive machine translation. Mach. Transl. 12, 175–194.
González-Rubio, J., Benedí, J.-M., Ortiz-Martínez, D., Casacuberta, F., 2016. Beyond prefix-based interactive translation prediction. In: Proceedings of the SIGNLL Conference on Computational Natural Language Learning. pp. 198–207.
González-Rubio, J., Ortiz-Martínez, D., Benedí, J.-M., Casacuberta, F., 2013. Interactive machine translation using hierarchical translation models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 244–254.
González-Rubio, J., Ortiz-Martínez, D., Casacuberta, F., 2010a. 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. pp. 173–177.
González-Rubio, J., Ortiz-Martínez, D., Casacuberta, F., 2010b. On the use of confidence measures within an interactive-predictive machine translation system. In: Proceedings of the Annual Conference of the European Association for Machine Translation.
González-Rubio, J., Ortiz-Martínez, D., Casacuberta, F., 2012. Active learning for interactive machine translation. In: Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics. pp. 245–254.
Goodfellow, I. J., Warde-Farley, D., Mirza, M., Courville, A., Bengio, Y., 2013. Maxout networks. In: Proceedings of the International Conference on Machine Learning. pp. 1319–1327.
Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., Schmidhuber, J., 2009. A novel connectionist system for unconstrained handwriting recognition. Inst. of Electr. and Electr. Eng. Trans. on Pattern Anal. and Mach. Intell. 31 (5), 855–868.
Graves, A., Mohamed, A.-r., Hinton, G., 2013. Speech recognition with deep recurrent neural networks. In: Proceedings of the Institute of Electrical and Electronics Engineers International Conference on Acoustics, Speech and Signal Processing. pp. 6645–6649.
Green, S., Chuang, J., Heer, J., Manning, C. D., 2014a. Predictive translation memory: A mixed-initiative system for human language translation. In: Proceedings of the Annual Association for Computing Machinery Symposium on User Interface Software and Technology. pp. 177–187.
Green, S., Wang, S., Chuang, J., Heer, J., Schuster, S., Manning, C. D., 2014b. Human effort and machine learnability in computer aided translation. In: Proceedings of the Empirical Methods in Natural Language Processing.
Gregor, K., Danihelka, I., Graves, A., Rezende, D., Wierstra, D., 2015. DRAW: A recurrent neural network for image generation. In: Proceedings of the International Conference on Machine Learning. pp. 1462–1471.
Gulcehre, C., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H.-C., Bougares, F., Schwenk, H., Bengio, Y., 2015. On using monolingual corpora in neural machine translation, `arXiv:1503.03535 [cs.CL]`.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.
Jean, S., Cho, K., Memisevic, R., Bengio, Y., 2015. On using very large target vocabulary for neural machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing. pp. 1–10.
Kalchbrenner, N., Blunsom, P., 2013. Recurrent continuous translation models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1700–1709.
Kneser, R., Ney, H., 1995. Improved backing-off for m-gram language modeling. In: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing. Vol. 1. IEEE, pp. 181–184.
Koehn, P., 2004. Statistical significance tests for machine translation evaluation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 388–395.
Koehn, P., 2005. Europarl: A parallel corpus for statistical machine translation. In: Proceedings of the Machine Translation Summit. pp. 79–86.
Koehn, P., 2010. A process study of computer-aided translation. Mach. Transl. 23 (4), 241–263.
Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E., 2007. Moses: Open source toolkit for statistical machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 177–180.
Koehn, P., Och, F. J., Marcu, D., 2003. Statistical phrase-based translation. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology. pp. 48–54.
Koehn, P., Tsoukala, C., Saint-Amand, H., 2014. Refinements to interactive translation prediction based on search graphs. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 574–578.
Langlais, P., Lapalme, G., 2002. TransType: Development-evaluation cycles to boost translator's productivity. Mach. Transl. 17 (2), 77–98.
Ling, W., Trancoso, I., Dyer, C., Black, A. W., 2015. Character-based neural machine translation, `arXiv:1511.04586 [cs.CL]`.
Liu, L., Utiyama, M., Finch, A., Sumita, E., 2016. Agreement on target-bidirectional neural machine translation. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics. pp. 411–416.
Luong, T., Pham, H., Manning, C. D., 2015a. Effective approaches to attention-based neural machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1412–1421.
Luong, T., Sutskever, I., Le, Q., Vinyals, O., Zaremba, W., 2015b. Addressing the rare word problem in neural machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing. pp. 11–19.
Macklovitch, E., 2006. TransType2: The last word. In: Proceedings of the International Conference on Language Resources and Evaluation. pp. 167–172.
Mathur, P., Cettolo, M., Federico, M., de Souza, J. G., 2014. Online multi-user adaptive statistical machine translation. In: Proceedings of the Association for Machine Translation in the Americas. pp. 152–165.
Mikolov, T., Karafiát, M., Burget, L., Cernockỳ, J., Khudanpur, S., 2010. Recurrent neural network based language model. In:

Proceedings of the Annual Conference of the International Speech Communication Association. pp. 1045–1048.
Nepveu, L., Lapalme, G., Langlais, P., Foster, G., 2004. Adaptive language and translation models for interactive machine translation. In: Proceedings of the Conference on Empirical Method in Natural Language Processing. pp. 190–197.
Nielsen, J., 1993. Usability Engineering. Morgan Kaufmann Publishers Inc.
Och, F. J., 2003. Minimum error rate training in statistical machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 160–167.
Och, F. J., Ney, H., 2002. Discriminative training and maximum entropy models for statistical machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 295–302.
Ortiz-Martínez, D., 2011. Advances in fully-automatic and interactive phrase-based statistical machine translation. Ph.D. thesis, Universidad Politécnica de Valencia, advisors: Ismael García Varea and Francisco Casacuberta.
Ortiz-Martínez, D., 2016. Online learning for statistical machine translation. Comput. Linguist. 42 (1), 121–161.
Papineni, K., Roukos, S., Ward, T., Zhu, W.-J., 2002. BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 311–318.
Peris, Á., Bolaños, M., Radeva, P., Casacuberta, F., 2016. Video description using bidirectional recurrent neural networks. In: Proceedings of the International Conference on Artificial Neural Networks. pp. 3–11.
Ranzato, M., Chopra, S., Auli, M., Zaremba, W., 2015. Sequence level training with recurrent neural networks, arXiv:1511.06732.
Sanchis-Trilles, G., Ortiz-Martínez, D., Casacuberta, F., 2014. Efficient wordgraph pruning for interactive translation prediction. In: Proceedings of the Annual Conference of the European Association for Machine Translation. pp. 27–34.
Sanchis-Trilles, G., Ortiz-Martínez, D., Civera, J., Casacuberta, F., Vidal, E., Hoang, H., 2008. Improving interactive machine translation via mouse actions. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 485–494.
Schwenk, H., 2007. Continuous space language models. Comput. Speech & Lang. 21 (3), 492–518.
Sennrich, R., Haddow, B., Birch, A., 2016. Neural machine translation of rare words with subword units. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 1715–1725.
Shen, S., Cheng, Y., He, Z., He, W., Wu, H., Sun, M., Liu, Y., 2015. Minimum risk training for neural machine translation, arXiv:1512.02433 [cs.CL].
Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J., 2006. A study of translation edit rate with targeted human annotation. In: Proceedings of the Association for Machine Translation in the Americas. pp. 223–231.
Stolcke, A., 2002. SRILM - an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing. pp. 901–904.
Sundermeyer, M., Alkhouli, T., Wuebker, J., Ney, H., 2014. Translation modeling with bidirectional recurrent neural networks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 14–25.
Sundermeyer, M., Schlüter, R., Ney, H., 2012. LSTM neural networks for language modeling. In: Proceedings of the Annual Conference of the International Speech Communication Association. pp. 194–197.
Sutskever, I., Vinyals, O., Le, Q. V., 2014. Sequence to sequence learning with neural networks. In: Proceedings of the Advances in Neural Information Processing Systems. Vol. 27. pp. 3104–3112.
Theano Development Team, 2016. Theano: A Python framework for fast computation of mathematical expressions, arXiv:1605.02688 [cs.SC].
Tiedemann, J., 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In: Recent Advances in Natural Language Processing. Vol. 5. pp. 237–248.
Tomás, J., Casacuberta, F., 2006. Statistical phrase-based models for interactive computer-assisted translation. In: Proceedings of the International Conference on Computational Linguistics/Association for Computational Linguistics. pp. 835–841.
Torregrosa, D., Forcada, M. L., Pérez-Ortiz, J. A., 2014. An open-source web-based tool for resource-agnostic interactive translation prediction. The Prague Bull. of Math. Linguist. 102, 69–80.
Vinyals, O., Kaiser, Ł., Koo, T., Petrov, S., Sutskever, I., Hinton, G., 2015a. Grammar as a foreign language. In: Advances in Neural Information Processing Systems. pp. 2755–2763.
Vinyals, O., Toshev, A., Bengio, S., Erhan, D., 2015b. Show and tell: A neural image caption generator. In: Proceedings of the Conference on Computer Vision and Pattern Recognition. pp. 3156–3164.
Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, Ł., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J., 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, arXiv:1609.08144 [cs.CL].
Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y., 2015. Show, attend and tell: Neural image caption generation with visual attention. In: Proceedings of the International Conference on Machine Learning. pp. 2048–2057.
Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A., 2015. Describing videos by exploiting temporal structure. In: Proceedings of the International Conference on Computer Vision. pp. 4507–4515.
Zaidan, O. F., Callison-Burch, C., 2010. Predicting human-targeted translation edit rate via untrained human annotators. In: Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics. pp. 369–372.
Zeiler, M. D., 2012. ADADELTA: An adaptive learning rate method, arXiv:1212.5701 [cs.LG].
Zens, R., Och, F. J., Ney, H., 2002. Phrase-based statistical machine translation. In: Proceedings of the Annual German Conference on Advances in Artificial Intelligence. Vol. 2479. pp. 18–32.