An Interactive Machine Translation Framework for Modernizing the Language of Historical Documents*

Miguel Domingo and Francisco Casacuberta

PRHLT Research Center
Universitat Politècnica de València
midobal@prhlt.upv.es, fcn@prhlt.upv.es

Abstract. In order to make historical documents accessible to a broader audience, language modernization generates a new version of a given document in the modern version of its original language. However, while they succeed in their goal, these modernizations are far from perfect. Thus, in order to help scholars generate error-free modernizations when the quality is essential, we propose an interactive language modernization framework based on interactive machine translation. In this work, we deployed two successful interactive protocols into language modernization. We evaluated our proposal on a simulated environment observing significant reductions of the human effort.

Keywords: Interactive Machine Translation · Language Modernization · Spelling Normalization · Historical Documents.

1 Introduction

Despite being an important part of our cultural heritage, historical documents are mostly limited to scholars. This is due to the nature of human language, which evolves with the passage of time, and the linguistic properties of these documents: the lack of spelling conventions made orthography to change depending on the author and time period. With the aim of making historical documents more accessible to non-experts, language modernization aims to automatically generate a new version of a given document written in the modern version of its original language.

However, language modernization is not error-free. While it succeeds in helping non-experts to understand the content of a historical document, there are times in which error-free modernized versions are needed. For example, scholars manually modernize the language of classic literature in order to make works understandable to contemporary readers [33]. In order to help scholars to generate these error-free versions, we propose to apply the interactive machine translation (IMT) framework into language modernization.

IMT proposes a collaborative framework in which a human and a translation system work together to produce the final translation. In this work, we propose to integrate two successful IMT protocols into language modernization. Our contributions are as follow:

* Author version of the paper published in *Proceedings of the Iberian Conference on Pattern Recognition and Image Analysis, 2022*. The final authenticated version is available online at https://doi.org/10.1007/978-3-031-04881-4_4.

- Integration of prefix-based and segment-based IMT into language modernization.
- Experimentation over three datasets from different languages and time periods.

2 Related work

Despite being manually applied to literature for centuries (e.g., *The Bible* has been adapted and translated for generations in order to preserve and transmit its contents [15]), automatic language modernization is a young research field. One of the first related works was a shared task for translating historical text to contemporary language [41]. While the task was focused on normalizing the document's spelling, they also approached language modernization using a set of rules. Domingo et al. [9] proposed a language modernization approach based on statistical machine translation (SMT). Domingo and Casacuberta [7] proposed a neural machine translation (NMT) approach. Sen et al. [35] augmented the training data by extracting pairs of phrases and adding them as new training sentences. Domingo and Casacuberta [8] proposed a method to profit from modern documents to enrich the neural models and conducted a user study. Lastly, Peng et al. [26] proposed a method for generating modernized summaries of historical documents.

IMT was introduced during the *TransType* project [11] and was further developed during *TransType2* [3]. New contributions to IMT include developing new generations of the suffix [43]; and profiting from the use of the mouse [34]. Marie et al. [22] introduced a touch-based interaction to iteratively improve translation quality. Lastly, Domingo et al. [10] introduced a segment-based protocol that broke the left-to-right limitation.

With the rise of NMT, the interactive framework was deployed into the neural systems [18,29], adding online learning techniques [28]; and reinforcement and imitation learning [20].

3 Language modernization approaches

Language modernization relies on machine translation (MT) which, given a source sentence x_1^J, aims to find the most likely translation $\hat{y}_1^{\hat{I}}$ [6]:

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

3.1 SMT approach

This approach relies on SMT, which has been the prevailing approach to compute Eq. (1) for years. SMT uses models that rely on a log-linear combination of different models [24]: namely, phrase-based alignment models, reordering models and language models; among others [45].

The SMT modernization approach tackles language modernization as a conventional translation task: given a parallel corpora—where for each original sentence of a document its modernized version is also available—an SMT system is trained by considering the language of the original document as the source language and its modernized version as the target language.

3.2 NMT approaches

These approaches are based on NMT, which models Eq. (1) with a neural network. While other architectures are possible, its most frequent architecture is based on an encoder-decoder, featuring recurrent networks [2,39], convolutional networks [13] or attention mechanisms [44]. The source sentence is projected into a distributed representation at the encoding state. Then, the decoder generates at the decoding step its most likely translation—word by word—using a beam search method [39]. The model parameters are typically estimated jointly—via stochastic gradient descent [32]—on large parallel corpora. Finally, at decoding time, the system obtains the most likely translation by means of a beam search method.

Like the previous approach, the NMT approaches tackle language modernization as a conventional translation task but using NMT instead of SMT. Moreover, due to NMT needing larger quantities of parallel training data than we have available (the scarce availability of parallel training data is a frequent problem for historical data [5]), we followed Domingo and Casacuberta's [8] proposal for enriching the neural models with synthetic data: Given a monolingual corpus, we apply feature decay algorithm (FDA) [4] to filter it and obtain a more relevant subset. Then, following a backtranslation approach [36], we train an inverse SMT system—using the modernized version of the training dataset as source, and the original version as target. After that, we translate the monolingual data with this system, obtaining a new version of the documents which, together with the original modern documents, conform the synthetic parallel data. Following that, we train an NMT modernization system with the synthetic corpus. Finally, we fine-tune the system by training a few more steps using the original training data.

We made use of two different NMT modernization approaches, whose difference is the architecture of the neural systems:

- NMT_{LSTM}: this approach uses a recurrent neural network (RNN) [16] architecture with long short-term memory (LSTM) [16] cells.
- $\text{NMT}_{\text{Transformer}}$: this approach uses a Transformer [44] architecture.

4 Interactive machine translation

In this work, we made use of two different IMT protocols: prefix-based and segment-based.

4.1 Prefix-based IMT

In this protocol, the system proposes an initial translation y_1^I of length I. Then, the user reviews it and corrects the leftmost wrong word y_i. Inherently, this correction validates all the words that precede this correction, forming a validated prefix $\tilde{y}_1^i$, that includes the corrected word $\tilde{y}_i$. Immediately, the system reacts to this user feedback ($f = \tilde{y}_1^i$), generating a suffix $\hat{y}_{i+1}^I$ that completes $\tilde{y}_1^i$ to obtain a new translation of $x_1^J : \hat{y}_i^I = \tilde{y}_1^i \, \hat{y}_{i+1}^I$. This process is repeated until the user accepts the system's complete suggestion.

The suffix generation was formalized by Barrachina et al. [3] as follows:

$$\hat{y}_{i+1}^{I} = \underset{I, y_{i+1}^{I}}{\arg\max}\, Pr(\tilde{y}_1^i \, y_{i+1}^I \mid x_1^J) \tag{2}$$

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

Similarly, Peris et al. [29] formalized the neural equivalent as follows:

$$p(\hat{y_{i'}} \mid \hat{y}_1^{i'-1}, x_1^J, f = \tilde{y}_1^i; \Theta) = \begin{cases} \delta(\hat{y_{i'}}, \tilde{y}_{i'}), & \text{if } i' \leq i \\ \bar{\mathbf{y}}_{i'}^{\top} \mathbf{p}_{i'} & \text{otherwise} \end{cases} \tag{3}$$

where x_1^J is the source sentence; $\tilde{y}_1^i$ is the validated prefix together with the corrected word; Θ are the models parameters; $\bar{\mathbf{y}}_{i'}^{\top}$ is the one hot codification of the word i'; $\mathbf{p}_{i'}$ contains the probability distribution produced by the model at time-step i; and $\delta(\cdot,\cdot)$ is the Kronecker delta.

This is equivalent to a forced decoding strategy and can be seen as generating the most probable suffix given a validated prefix, which fits into the statistical framework deployed by Barrachina et al. [3].

4.2 Segment-based IMT

This protocol extends the human–computer collaboration from the prefix-based protocol. Now, besides correcting a word, the user can validate segments (sequences of words) and combine consecutive segments to create a larger one.

Like with the previous protocol, the process starts with the system suggesting an initial translation. Then, the user reviews it and validates those sequences of words which they consider to be correct. After that, they can delete words between validated segments to create a larger segment. Finally, they make a word correction.

These three actions constitute the user feedback, which Domingo et al. [10] formalized as: $\tilde{\mathbf{f}}_1^N = \tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N$; where $\tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N$ is the sequence of N correct segments validated by the user in an interaction. Each segment is defined as a sequence of one or more target words, so each action taken by the user modifies the feedback in a different way. Therefore, the user can:

1. Validate a new segment, inserting a new segment $\tilde{\mathbf{f}}_i$ in $\tilde{\mathbf{f}}_1^N$.
2. Merge two consecutive segments $\tilde{\mathbf{f}}_i$, $\tilde{\mathbf{f}}_{i+1}$ into a new one.
3. Introduce a word correction. This is introduced as a new one-word validated segment, $\tilde{\mathbf{f}}_i$, which is inserted in $\tilde{\mathbf{f}}_1^N$.

The first two actions are optional (an iteration may not have new segments to validate) while the last action is mandatory: it triggers the system to react to the user feedback, starting a new iteration of the process.

The system's reaction to the user's feedback results in a sequence of new translation segments $\widehat{\mathbf{h}}_0^{N+1} = \widehat{\mathbf{h}}_0, \ldots, \widehat{\mathbf{h}}_{N+1}$. That means, an $\widehat{\mathbf{h}}_i$ for each pair of validated segments $\tilde{\mathbf{f}}_i, \tilde{\mathbf{f}}_{i+1}$, being $1 \leq i \leq N$; plus one more at the beginning of the hypothesis, $\widehat{\mathbf{h}}_0$; and another at the end of the hypothesis, $\widehat{\mathbf{h}}_{N+1}$. The new translation of x_1^J is obtained by alternating validated and non-validated segments: $\hat{y}_1^I = \widehat{\mathbf{h}}_0, \tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N, \widehat{\mathbf{h}}_{N+1}$. The goal is to obtain the best sequence of translation segments, given the user's feedback and the source sentence:

$$\widehat{\mathbf{h}}_0^{N+1} = \underset{\mathbf{h}_0^{N+1}}{\arg\max}\, Pr(\mathbf{h}_0, \tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N, \mathbf{h}_{N+1} \mid x_1^J) \tag{4}$$

This equation is very similar to Eq. (2). The difference is that, now, the search is performed in the space of possible substrings of the translations of x_1^J, constrained by the sequence of segments $\tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N$, instead of being limited to the space of suffixes constrained by $\tilde{y}_1^i$.

Similarly, Peris et al. [29] formalized the neural equivalent of this protocol as follows:

$$p(y_{i_n+i'} \mid y_1^{i_n+i'-1}, x_1^J, f_1^N; \Theta) = \mathbf{y}_{i_n+i'}^\top \mathbf{p}_{i_n+i'} \tag{5}$$

where $f_1^N = f_1, \ldots, f_N$ is the feedback signal and $f_1, \ldots, f_N$ are a sequence of non-overlapping segments validated by the user; each alternative hypothesis y (partially) has the form $y = \ldots, f_n, h_n, f_{n+i}, \ldots$; g_n is the non-validated segment; $1 \leq i' \leq \hat{l}_n$; and l_n is the size of this non-validated segment and is computed as follows:

$$\hat{l}_n = \underset{0 \leq l_n \leq L}{\arg\max} \frac{1}{l_N + 1} \sum_{i'=i_n+1}^{i_n+l_n+1} \log p(y_{i'} \mid y_1^{i'-1}, x_1^J; \Theta) \tag{6}$$

5 Experimental framework

In this section, we present the details of our experimental session: user simulation, systems, corpora and metrics.

5.1 User simulation

In this work, we conducted an evaluation with simulated users due to the time and economic costs of conducting frequent human evaluations during the development stage. These users had as goal to generate the modernizations from the reference.

Prefix-based simulation At each iteration, the user compares the system's hypothesis with the reference and looks for the leftmost different word. When they find it, they make a correction, validating a new prefix in the process. This correction has an associated cost of one mouse action and one word stroke. The system, then, reacts to the user feedback and generates a new suffix that completes the prefix to conform a new modernization hypothesis. This process is repeated until the hypothesis and the reference are the same.

This simulation was conducted using Domingo et al.'s [10] updated version of Barrachina et al.'s [3] software[1] for the SMT systems, and *NMT-Keras* [27]'s interactive branch for the NMT systems.

[1] `https://github.com/midobal/pb-imt`.

Segment-based simulation Like Domingo et al. [10], in this simulation we assumed that validated word segments must be in the same order as in the reference. Therefore, segments that need to be reordered are not validated. Moreover, for the sake of simplicity and without loss of generality, we assumed that the user always corrects the leftmost wrong word.

At each iteration, the user validates segments by computing the longest common subsequence [1] between the system's hypothesis and the reference. This has an associated cost of one action for each one-word segment and two actions for each multi-word segment. Then, the user checks if any pair of consecutive validated segments should be merged into a single segment (i.e., they appear one consecutively in the reference but are separated by some words in the hypothesis). If so, they merge them, increasing mouse actions in one where there is a single word between them or two otherwise. Finally, they correct the leftmost wrong word (like in the prefix-based simulation). The system, then, reacts to this feedback generating a new hypothesis. This process is repeated until the hypothesis and the reference are the same.

This simulation was conducted using Domingo et al.'s [10] software[2] for the SMT systems, and *NMT-Keras*'s [27] interactive branch for the NMT systems.

5.2 Systems

SMT systems were trained with *Moses* [19], following the standard procedure: we estimated a 5-gram language model—smoothed with the improved KneserNey method—using *SRILM* [38], and optimized the weights of the log-linear model with MERT [23]. SMT systems were used both for the SMT modernization approach and for generating synthetic data to enrich the neural systems (see Section 3.2).

We built NMT systems using *NMT-Keras* [27]. We used long short-term memory units [14], with all model dimensions set to 512 for the RNN architecture. We trained the system using Adam [17] with a fixed learning rate of 0.0002 and a batch size of 60. We applied label smoothing of 0.1 [40]. At inference time, we used beam search with a beam size of 6. In order to reduce vocabulary, we applied joint byte pair encoding (BPE) [12] to all corpora, using $32,000$ merge operations.

For the Transformer architecture [44], we used 6 layers; Transformer, with all dimensions set to 512 except for the hidden Transformer feed-forward (which was set to 2048); 8 heads of Transformer self-attention; 2 batches of words in a sequence to run the generator on in parallel; a dropout of 0.1; Adam [17], using an Adam beta2 of 0.998, a learning rate of 2 and Noam learning rate decay with 8000 warm up steps; label smoothing of 0.1 [40]; beam search with a beam size of 6; and joint BPE applied to all corpora, using $32,000$ merge operations.

5.3 Corpora

We made use of the following corpora in our experimental session:

[2] `https://github.com/midobal/sb-imt`.

Dutch Bible [41]: A collection of different versions of the Dutch Bible. Among others, it contains a version from 1637—which we consider as the original version—and another from 1888—which we consider as the modern version (using 19$^{\text{th}}$ century Dutch as if it were *modern Dutch*).

El Quijote [7]: the well-known 17$^{\text{th}}$ century Spanish novel by Miguel de Cervantes, and its correspondent 21$^{\text{st}}$ century version.

OE-ME [35]: contains the original 11$^{\text{th}}$ century English text *The Homilies of the Anglo-Saxon Church* and a 19$^{\text{th}}$ century version—which we consider as *modern English*.

Additionally, to enrich the neural models we made use of the following *modern documents*: the collection of Dutch books available at the *Digitale Bibliotheek voor de Nederlandse letteren*[3], for Dutch; and OpenSubtitles [21]—a collection of movie subtitles in different languages—for Spanish and English. Table 1 contains the corpora statistics.

		Dutch Bible		El Quijote		OE-ME	
		Original	**Modernized**	**Original**	**Modernized**	**Original**	**Modernized**
Train	$\|S\|$	35.2K		10K		2716	
	$\|T\|$	870.4K	862.4K	283.3K	283.2K	64.3K	69.6K
	$\|V\|$	53.8K	42.8K	31.7K	31.3K	13.3K	8.6K
Validation	$\|S\|$	2000		2000		500	
	$\|T\|$	56.4K	54.8K	53.2K	53.2K	12.2K	13.3K
	$\|V\|$	9.1K	7.8K	10.7K	10.6K	4.2K	3.2K
Test	$\|S\|$	5000		2000		500	
	$\|T\|$	145.8K	140.8K	41.8K	42.0K	11.9K	12.9K
	$\|V\|$	10.5K	9.0K	8.9K	9.0K	4.1K	3.2K
Modern documents	$\|S\|$	3.0M		2.0M		6.0M	
	$\|T\|$	76.1M	74.1M	22.3M	22.2M	67.5M	71.6M
	$\|V\|$	1.7M	1.7M	210.1K	211.7K	290.2K	287.4K

Table 1. Corpora statistics. $|S|$ stands for number of sentences, $|T|$ for number of tokens and $|V|$ for size of the vocabulary. *Modern documents* refers to the monolingual data used to create the synthetic data. M denotes millions and K thousands.

5.4 Metrics

In order to assess our proposal, we make use of the following well-known metrics:

Word Stroke Ratio (WSR) [42]: measures the number of words edited by the user, normalized by the number of words in the final translation.

Mouse Action Ratio (MAR) [3]: measures the number of mouse actions made by the user, normalized by the number of characters in the final translation.

Additionally, to assess the initial quality of the modernization systems, we used the following well-known metrics:

[3] `http://dbnl.nl/`.

BiLingual Evaluation Understudy (BLEU) [25]: computes the geometric average of the modified n-gram precision, multiplied by a brevity factor that penalizes short sentences. In order to ensure consistent BLEU scores, we used *sacreBLEU* [30] for computing this metric.

Translation Error Rate (TER) [37]: computes the number of word edit operations (insertion, substitution, deletion and swapping), normalized by the number of words in the final translation.

Finally, we applied approximate randomization testing (ART) [31]—with $10,000$ repetitions and using a p-value of 0.05—to determine whether two systems presented statistically significance.

6 Results

Table 2 presents the experimental results. It presents the initial quality of each modernization system and compares their performance on a prefix-based or a segment-based framework.

Corpus	Approach	Modernization quality TER [↓]	Modernization quality BLEU [↑]	Prefix-based WSR [↓]	Prefix-based MAR [↓]	Segment-based WSR [↓]	Segment-based MAR [↓]
Dutch Bible	SMT	11.5	77.5	14.3	4.4	**9.0**	**10.8**
	NMT_{LSTM}	$50.7^{\dagger}$	43.4	$42.6^{\ddagger}$	9.2	$42.6^{\ddagger}$	50.9
	$\text{NMT}_{\text{Transformer}}$	$50.3^{\dagger}$	35.8	$49.2^{\ddagger}$	10.4	$49.2^{\ddagger}$	48.3
El Quijote	SMT	30.7	58.3	38.8	10.9	**22.0**	**19.7**
	NMT_{LSTM}	42.9	50.4	$68.9^{\ddagger}$	11.8	$68.9^{\ddagger}$	47.8
	$\text{NMT}_{\text{Transformer}}$	47.3	46.1	$73.2^{\ddagger}$	13.4	$73.2^{\ddagger}$	50.5
OE-ME	SMT	39.6	39.6	58.2	15.5	**28.2**	**26.1**
	NMT_{LSTM}	56.4	30.3	$72.1^{\ddagger}$	$12.8^{\dagger}$	$72.1^{\ddagger}$	59.5
	$\text{NMT}_{\text{Transformer}}$	58.9	28.2	$73.5^{\ddagger}$	$13.3^{\dagger}$	$73.5^{\ddagger}$	49.5

Table 2. Experimental results. The initial modernization quality is meant to be a starting point comparison of each system. All results are significantly different between all approaches except those denoted with †. Given the same approach, all results are significantly different between the different IMT protocols except those denoted with ‡. [↓] indicates that the lowest the value the highest the quality. [↑] indicates that the highest the value the highest the quality. Best results are denoted in **bold**.

In all cases, the SMT approach yielded the best results with a great difference. The prefix-based protocol successfully reduces the human effort of creating error-free modernizations. Moreover, the segment-based protocol reduced the typing effort even more, at the expenses of a small increase in the use of the mouse, which is believed to have a smaller impact in the human effort [10].

With respect to the NMT approaches, while all of them also successfully decreased the human effort, these diminishes are significantly smaller than with the SMT approach. Moreover, the segment-based protocol does not offer any benefit with respect to the prefix-based: both protocols have the same typing effort. However, the segment-based protocol has a significant increase in the mouse usage. Finally, is it worth noting how the initial quality of the systems is considerably lower than the SMT approach, which has an impact in the IMT performance.

source ($\mathbf{x}$)**:** Ealle ðing he foresceawað and wát, and ealra ðeoda gereord he cann.
target translation ($\hat{\mathbf{y}}$)**:** All things he foresees and knows, and he understands the tongues of all nations.

IT-0	MT	All things he foresceawað and knows, and of all nations language he understands.
IT-1	User	All things he **foresees** and knows, and of all nations language he understands.
	MT	All things he foresees and knows, and of all nations language he understands.
IT-2	User	All things he foresees and knows, and **he** all nations language he understands.
	MT	All things he foresees and knows, and he understands of all nations language.
IT-3	User	All things he foresees and knows, and he understands **the** all nations language.
	MT	All things he foresees and knows, and he understands the beginning of all nations language.
IT-4	User	All things he foresees and knows, and he understands the **tongues** of all nations language.
	MT	All things he foresees and knows, and he understands the tongues all.
IT-5	User	All things he foresees and knows, and he understands the tongues **of**
	MT	All things he foresees and knows, and he understands the tongues of all.
IT-6	User	All things he foresees and knows, and he understands the tongues of all **nations.**
	MT	All things he foresees and knows, and he understands the tongues of all nations .
END	User	All things he foresees and knows, and he understands the tongues of all nations.

Table 3. Example of a prefix-based IMT session. The session starts with the system proposing an initial modernization. The user, then, looks for the leftmost wrong word and corrects it (*foresees* instead of *foresceawað*). Inherently, they are validating the prefix *All thing he*. Immediately, the system reacts to this feedback by suggesting a new hypothesis. The process is repeated until the user finds the system's hypothesis satisfactory.

6.1 Quality analysis

Finally, we show some examples which reflect the benefits of using an interactive framework. Table 3 showcases a prefix-based IMT session for modernazing the language of a sentence from an old English document. Modernizing the sentence from scratch has an associated cost of 14 word strokes and one mouse action, while correcting the automatic modernization has an associated cost of 7 word strokes and 7 mouse actions. However, with the prefix-based protocol this cost is reduced to 6 word strokes and 6 mouse actions. Furthermore, with the segment-based protocol (see Table 4) this cost is reduced to only 3 word strokes at the expenses of increasing the mouse actions—which have a smaller impact in the human effort—to 15.

7 Conclusions and future work

In this work we have deployed the interactive framework into language modernization in order to help scholar generate error-free modernizations. We deployed two different protocols to SMT and NMT modernization approaches.

Under simulated conditions, we observed that, while the IMT framework always succeeded in reducing the human effort, the SMT approach yielded the best results. Moreover, while the segment-based protocol performed significantly better than the

source (x): Ealle ðing he foresceawað and wát, and ealra ðeoda gereord he cann.

target translation ($\hat{\mathbf{y}}$): All things he foresees and knows, and he understands the tongues of all nations.

IT-0	MT	All things he foresceawað and knows, and of all nations language he understands.
IT-1	User	[All things he] **foresees** [and knows,] [and] [of all nations] language he understands.
	MT	[All things he] foresceawað [foresees] [and knows,] [and] language he understand [of all nations] .
IT-2	User	[All things he foresees and knows, and he understand] **the** [of all nations] .
	MT	[All things he foresees and knows, and he understand] [the] language [of all nations] .
IT-3	User	[All things he foresees and knows, and he understand the] **tongues** [of all nations] .
	MT	[All things he foresees and knows, and he understand the] [tongues] [of all nations] .
END	User	All things he foresees and knows, and he understands the tongues of all nations.

Table 4. Example of a segment-based IMT session. The session starts with the system proposing an initial modernization. The user, then, reviews the hypothesis and selects all the word segments that considers to be correct ([All things he] , [and knows] , [and] and [of all nations]). Then they make a word correction (*foresees* instead of *foresceawað*). Immediately, the system reacts to this feedback by suggesting a new hypothesis. The process is repeated until the user finds the system's hypothesis satisfactory.

prefix-based protocol for the SMT approach, there was no significantly difference for the NMT approaches.

Finally, in a future work we would like to conduct a human evaluation with the help of scholars to better assess the benefits of the interactive language modernization framework.

Acknowledgements

The research leading to these results has received funding from *Generalitat Valenciana* under project *PROMETEO/2019/121*. We gratefully acknowledge *Andrés Trapiello* and *Ediciones Destino* for granting us permission to use their book in our research.

References

1. Apostolico, A., Guerra, C.: The longest common subsequence problem revisited. Algorithmica **2**, 315–336 (1987)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2015), *arXiv:1409.0473*
3. Barrachina, S., Bender, O., Casacuberta, F., Civera, J., Cubel, E., Khadivi, S., Lagarda, A., Ney, H., Tomás, J., Vidal, E., Vilar, J.M.: Statistical approaches to computer-assisted translation. Computational Linguistics **35**, 3–28 (2009)
4. Biçici, E., Yuret, D.: Optimizing instance selection for statistical machine translation with feature decay algorithms. IEEE/ACM Transactions on Audio, Speech and Language Processing **23**(2), 339–350 (2015)
5. Bollmann, M., Søgaard, A.: Improving historical spelling normalization with bi-directional lstms and multi-task learning. In: Proceedings of the International Conference on the Computational Linguistics. pp. 131–139 (2016)
6. Brown, P.F., Pietra, V.J.D., Pietra, S.A.D., Mercer, R.L.: The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics **19**(2), 263–311 (1993)

7. Domingo, M., Casacuberta, F.: A machine translation approach for modernizing historical documents using back translation. In: Proceedings of the International Workshop on Spoken Language Translation. pp. 39–47 (2018)
8. Domingo, M., Casacuberta, F.: Modernizing historical documents: A user study. Pattern Recognition Letters **133**, 151–157 (2020). https://doi.org/10.1016/j.patrec.2020.02.027
9. Domingo, M., Chinea-Rios, M., Casacuberta, F.: Historical documents modernization. The Prague Bulletin of Mathematical Linguistics **108**, 295–306 (2017)
10. Domingo, M., Peris, Á., Casacuberta, F.: Segment-based interactive-predictive machine translation. Machine Translation **31**, 1–23 (2017)
11. Foster, G., Isabelle, P., Plamondon, P.: Target-text mediated interactive machine translation. Machine Translation **12**, 175–194 (1997)
12. Gage, P.: A new algorithm for data compression. The C Users Journal **12**(2), 23–38 (1994)
13. Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N.: Convolutional sequence to sequence learning. In: Proceedings of the International Conference on Machine Learning. p. 1243–1252 (2017)
14. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: Continual prediction with LSTM. Neural computation **12**(10), 2451–2471 (2000)
15. Given, M.D.: A discussion of bible translations and biblical scholarship. http://courses.missouristate.edu/markgiven/rel102/bt.htm (2015)
16. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
18. Knowles, R., Koehn, P.: Neural interactive translation prediction. In: Proceedings of the Association for Machine Translation in the Americas. pp. 107–120 (2016)
19. 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.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 177–180 (2007)
20. Lam, T.K., Schamoni, S., Riezler, S.: Interactive-predictive neural machine translation through reinforcement and imitation. In: Proceedings of Machine Translation Summit. pp. 96–106 (2019)
21. Lison, P., Tiedemann, J.: Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In: Proceedings of the International Conference on Language Resources Association. pp. 923–929 (2016)
22. Marie, B., Max, A.: Touch-based pre-post-editing of machine translation output. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1040–1045 (2015)
23. Och, F.J.: Minimum error rate training in statistical machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 160–167 (2003)
24. Och, F.J., Ney, H.: 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 (2002)
25. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 311–318 (2002)
26. Peng, X., Zheng, Y., Lin, C., Siddharthan, A.: Summarising historical text in modern languages. In: Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics. pp. 3123–3142 (2021)

27. Peris, A., Casacuberta, F.: NMT-Keras: a Very Flexible Toolkit with a Focus on Interactive NMT and Online Learning. The Prague Bulletin of Mathematical Linguistics **111**, 113–124 (2018)
28. Peris, Á., Casacuberta, F.: Online learning for effort reduction in interactive neural machine translation. Computer Speech & Language **58**, 98–126 (2019)
29. Peris, Á., Domingo, M., Casacuberta, F.: Interactive neural machine translation. Computer Speech & Language **45**, 201–220 (2017)
30. Post, M.: A call for clarity in reporting bleu scores. In: Proceedings of the Third Conference on Machine Translation. pp. 186–191 (2018)
31. Riezler, S., Maxwell, J.T.: On some pitfalls in automatic evaluation and significance testing for mt. In: Proceedings of the workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. pp. 57–64 (2005)
32. Robbins, H., Monro, S.: A stochastic approximation method. The Annals of Mathematical Statistics pp. 400–407 (1951)
33. Rodríguez Marcos, J.: Un 'Quijote' moderno. `https://elpais.com/cultura/2015/05/27/babelia/1432726379_211033.html` (2015)
34. Sanchis-Trilles, G., Ortiz-Martínez, D., Civera, J., Casacuberta, F., Vidal, E., Hoang, H.: Improving interactive machine translation via mouse actions. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 485–494 (2008)
35. Sen, S., Hasanuzzaman, M., Ekbal, A., Bhattacharyya, P., Way, A.: Take help from elder brother: Old to modern english nmt with phrase pair feedback. In: Proceedings of the International Conference on Computational Linguistics and Intelligent Text Processing (2019), in press
36. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. pp. 1715–1725 (2016)
37. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A study of translation edit rate with targeted human annotation. In: Proceedings of the Association for Machine Translation in the Americas. pp. 223–231 (2006)
38. Stolcke, A.: SRILM - an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing. pp. 257–286 (2002)
39. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Proceedings of the Advances in Neural Information Processing Systems. vol. 27, pp. 3104–3112 (2014)
40. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–9 (2015)
41. Tjong Kim Sang, E., Bollmann, M., Boschker, R., Casacuberta, F., Dietz, F., Dipper, S., Domingo, M., van der Goot, R., van Koppen, M., Ljubešić, N., Östling, R., Petran, F., Pettersson, E., Scherrer, Y., Schraagen, M., Sevens, L., Tiedemann, J., Vanallemeersch, T., Zervanou, K.: The CLIN27 shared task: Translating historical text to contemporary language for improving automatic linguistic annotation. Computational Linguistics in the Netherlands Journal **7**, 53–64 (2017)
42. Tomás, J., Casacuberta, F.: 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 (2006)
43. Torregrosa, D., Forcada, M.L., Pérez-Ortiz, J.A.: An open-source web-based tool for resource-agnostic interactive translation prediction. Prague Bulletin of Mathematical Linguistics **102**, 69–80 (2014)

44. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. pp. 5998–6008 (2017)
45. Zens, R., Och, F.J., Ney, H.: Phrase-based statistical machine translation. In: Proceedings of the Annual German Conference on Advances in Artificial Intelligence. vol. 2479, pp. 18–32 (2002)