
Article:
Boito, M.Z., Villavicencio, A. orcid.org/0000-0002-3731-9168 and Besacier, L. (2021) Investigating alignment interpretability for low-resource NMT. Machine Translation, 34. pp. 305-323. ISSN 0922-6567

https://doi.org/10.1007/s10590-020-09254-w

Noname manuscript No.
(will be inserted by the editor)

Investigating Alignment Interpretability for Low-resource NMT

Marcely Zanon Boito · Aline Villavicencio · Laurent Besacier

Received: date / Accepted: date

Abstract The attention mechanism for Neural Machine Translation (NMT) added flexibility to neural models, and the possibility to visualize *soft-alignments* between source and target representations. While there is much debate about the impact of attention in the translation quality of neural models [25, 40, 35, 32], in this paper we propose a different assessment, investigating soft-alignment interpretability in low-resource scenarios. We experiment with different architectures (RNN [5], 2D-CNN [15], and Transformer [36]), comparing their capacity to produce directly exploitable alignments. For evaluating exploitability, we replicate the Unsupervised Word Segmentation (UWS) task from Godard et al. [21]. There, source words are translated into unsegmented phone sequences. Posterior to training, the resulting soft-alignments are used for producing segmentation over the target side. Our results show that the RNN produces the most exploitable alignments in this scenario. We thus conclude by investigating methods for increasing its UWS scores. We compare the following methodologies: monolingual pre-training, input representation augmentation (hybrid model), and explicit word length optimization during training. We reach the best results by using the hybrid model, which uses an intermediate monolingual-rooted segmentation from a non-parametric Bayesian model [24] to enrich the input representation before training.

Keywords low-resource languages · attention mechanism · sequence-to-sequence models · unsupervised word segmentation · computational language documentation · neural machine translation

Marcely Zanon Boito and Laurent Besacier
University Grenoble Alpes: Laboratoire d'Informatique de Grenoble (France)
E-mail: marcely.zanon-boito@univ-grenoble-alpes.fr

Aline Villavicencio
University of Sheffield: Department of Computer Science (England)
Federal Univesity of Rio Grande do Sul: Institute of Informatics (Brazil)

1 Introduction

Recently, encoder-decoder architectures equipped with attention mechanisms emerged as a popular solution for addressing sequence-to-sequence (S2S) problems for a variety of tasks. These include Automatic Speech Recognition [39, 11], Text-to-Speech Synthesis [38,33], and Neural Machine Translation (NMT) [5,15,36,17,34]. For NMT, popular leaderboards, such as WMT 2014 and IWSLT 2015, have been dominated by these attention-based approaches for years now.[1]

An interesting effect of encoder-decoder attention is the possibility of visualizing *soft-alignment* between source and target sentences posterior to training. Several NMT architecture papers provide these visualizations as a way of attesting training quality, relating the soft-alignment with bilingual alignments. However, recent studies focused on the interpretability of attention mechanisms failed to find a strong connection between the soft-alignment matrices' quality and the systems' performance [25,40,35,32]. While this does not mean the produced soft-alignments are meaningless, these studies highlight that they will not always directly relate to, or impact the word ranking obtained during decoding stage. Consequently, the soft-alignments should not be considered as directly responsible for the model's translation capacities, and instead should be seen as a by-product of training NMT models.

Nonetheless, the existence of an unsupervised source-to-target alignment mechanism inside NMT models remains a useful resource. Recent works [13, 21,10] trained NMT models between speech and translation sentences, and used the soft-alignment weights for performing word segmentation at speech-level. These studies also highlight that the attention mechanism is robust to low-resource settings, resting exploitable even when only few parallel sentences (5k) were available for training. On this line of work, Zenkel et al. [41] scored word-level alignment obtained through NMT training in well-resource settings, and Garg et al. [16] and Godard et al. [20] are examples of works that perform explicit optimization of the attention layer's soft-alignments, weighting the quality of the discovered alignments during training.

In this paper we investigate alignment interpretability in low-resource settings extending the work of Boito et al. [10]. We compare three well-known architectures for attention-based NMT with respect to their capability of generating directly interpretable alignments. These three architectures are: 2D Convolutional Neural Networks (2D-CNN) [15], Recurrent Neural Networks (RNN) [5], and Transformer [36]. For evaluating the generated alignments, we model the task following Godard et al. [21]. There, NMT models are trained for translating words in a source language into phone sequences in a target language, inferring from it word segmentation over the target side. This comes from the hypothesis that, if a model is able to generate a directly exploitable alignment between source and target, it will naturally produce

[1] Leaderboards available at: `https://paperswithcode.com/task/machine-translation`

word-level segmentation when translating words into phones.[2] Finally, while this Unsupervised Word Segmentation (UWS) task can be used for the extrinsic evaluation of the soft-alignment weights obtained, we also measure Average Normalized Entropy (ANE), a task-agnostic confidence metric to quantify the quality of the source-to-target alignments.

During this architecture comparison, we find that RNN produce the most exploitable alignments for the UWS task. We thus follow this investigation by studying methods for increasing the quality of this architecture's generated alignments. We do so by considering the UWS problem in the light of Computational Language Documentation (CLD). CLD is an emerging field [1,2,8, 28,6], whose main goal is the creation of automatic approaches able to help the documentation of the many languages soon to be extinct [3].

The first method investigated consists of pre-training the neural models on a smaller monolingual, manually annotated, subset can bias the attention mechanism towards better segmentation. We follow this by investigating if *boundary clues* inserted into the unsegmented phone sequence could enrich the representation learned by the decoder network. These boundary clues are extracted from a Bayesian segmentation system.[3] Finally, we investigate an explicit word-length optimization training method initially introduced by Godard et al. [20], and its impact on alignment quality.

All our models are trained in realistic documentation settings (only 5,130 aligned sentences), and evaluated considering their performance segmenting a true unwritten language: Mboshi (Bantu C25) [19]. Experiments confirm that all three methods, at different degrees, result in better exploitable alignments for the RNN model. Nonetheless, adding boundary clues into the input representation provides the best segmentation improvement over the baseline. This hints that intermediate annotations made by linguists during documentation could be leveraged during training.

Summarizing, this paper studies the interpretability of the soft-alignments produced by NMT architectures in low-resource settings. We first present our evaluation methodology (Section 2), followed by the description of the architectures investigated (Section 3). We compare them for the UWS task (Section 4), and study methods for increasing the quality of the produced alignments (Section 5). Section 6 concludes this work.

[2] The translation direction matters, since the attention layer outputs probability distribution over the encoder annotations for every decoder symbol generated. Translating in the opposite direction may result in phones being ignored [9].

[3] They could correspond, in another kind of scenario, to intermediate word boundaries developed by linguists during language documentation.

2 Alignment Assessment Methodology

2.1 Unsupervised Word Segmentation from Speech

Godard et al. [21] introduced a pipeline for performing Unsupervised Word Segmentation (UWS) from speech. The system outputs time-stamps delimiting stretches of speech, associated with class labels, corresponding to real words in the language. The pipeline consists of first transforming speech into a sequence of phones, either through Automatic Unit Discovery (e.g. [30]) or manual transcription. The phone sequences, together with their translations, are then fed to an attention-based S2S system that produces soft-alignment probability matrices between target and source languages. The alignment probability distributions between the phones and the translation words (as in Figure 1) are used to cluster (segment) together neighbor phones whose alignment distribution peaks at the same source word, as, for example, the phones phn25-phn10-phn60-phn10 and the word *monzo* in the first matrix from Figure 1. The final speech segmentation is evaluated using the *Zero Resource Challenge*[4] (ZRC) 2017 [12] evaluation suite (track 2).

2.2 Average Normalized Entropy

To assess the overall quality of the soft-alignment probability matrices without having gold alignment information, we use Average Normalized Entropy (ANE) [10]. Given the source and target pair $(\boldsymbol{s}, \boldsymbol{t})$ of lengths $|\boldsymbol{s}|$ and $|\boldsymbol{t}|$ respectively, for every phone t_i, the normalized entropy (NE) is computed considering all possible words in s (Eq. 1), where $P(t_i, s_j)$ is the alignment probability between the phone t_i and the word s_j (a cell in the matrix). The ANE for a sentence is then defined by the arithmetic mean over the resulting NE for every phone from the sequence t (Eq. 2). From this definition, we can derive ANE for different granularities (sub or supra-sentential) by accumulating its value for the full corpus, for a single type or for a single token. *Corpus ANE* will be used to summarize the overall performance of a S2S model on a specific corpus. *Token ANE* extends ANE to tokens by averaging NE for all phones from a single (discovered) token. *Type ANE* results from averaging the ANE for every token instance of a discovered type. Finally, *Alignment ANE* is the result of averaging the ANE for every discovered *(type, translation word)* alignment pair. The intuition that lower ANEs correspond to better alignments is exemplified in Figure 1.

$$NE(t_i, s) = -\sum_{j=1}^{|\boldsymbol{s}|} P(t_i, s_j) \cdot \log_{|\boldsymbol{s}|}(P(t_i, s_j)) \tag{1}$$

$$ANE(t, s) = \frac{\sum_{i=1}^{|\boldsymbol{t}|} NE(t_i, s)}{|\boldsymbol{t}|} \tag{2}$$

[4] Available at `http://zerospeech.com/2017`.

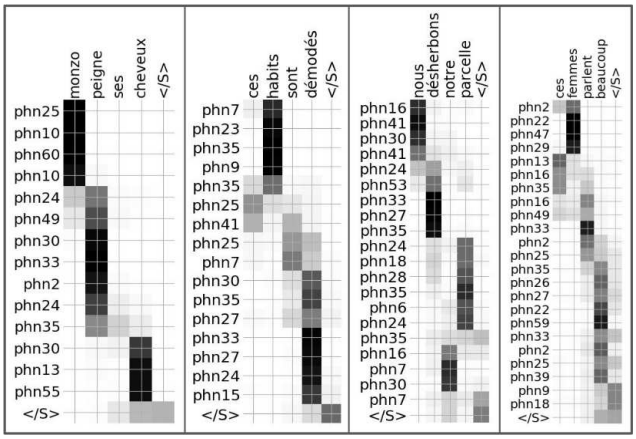

Fig. 1 Soft-alignment probability matrices from the alignment complexity buckets 1 to 4 (left to right) for examples with same source length. Darker squares correspond to higher probabilities. The sentence ANE scores are, from left to right, 0.26, 0.40, 0.47 and 0.53. The language pair is French (words) to Mboshi (phones).

3 Attention-based NMT Architectures

3.1 RNN: Encoder-Decoder Attention

The classic RNN encoder-decoder model [5] connects a bidirectional encoder with an unidirectional decoder by the use of an *alignment module*. The RNN encoder learns annotations for every source token, and these are weighted by the alignment module for the generation of every target token. Weights allow the computation of *context vectors*, capturing the *importance* of every source token for the generation of each target token.

Attention mechanism: a context vector for a decoder step t is computed using the set of source annotations H and the last state of the decoder network (translation context). The attention is the result of the weighted sum of the source annotations H (with $H = h_1, ..., h_{|s|}$) and their α probabilities (Eq. 3) obtained through a feed-forward network *align* (Eq. 4).

$$c_t = Att(H, s_{t-1}) = \sum_{j=1}^{|s|} \alpha_{t,j} h_j \tag{3}$$

$$\alpha_{t,j} = \text{softmax}(align(h_j, s_{t-1})) \tag{4}$$

3.2 Transformer: Multi-head Attention

Transformer [36] is a fully attentional S2S architecture, which has obtained state-of-the-art results for several NMT shared tasks. It replaces the use of

sequential cell units (such as LSTM) by Multi-Head Attention (MHA) operations, which make the architecture considerably faster. Both encoder and decoder networks are stacked layers sets that receive source and target sequences, embedded and concatenated with positional encoding. An encoder block is made of two sub-layers: a *Self-Attention* MHA and a feed-forward sub-layer. A decoder block is made of three sub-layers: a *masked Self-Attention* MHA (no access to subsequent positions); an *Encoder-Decoder* MHA (operation over the encoder stack's final output and the decoder self-attention output); and a feed-forward sub-layer. Dropout and residual connections are applied between all sub-layers. Final output probabilities are generated by applying a linear projection over the decoder stack's output, followed by a softmax operation.
Multi-head attention mechanism: attention is seen as a mapping problem in which, given a pair of key-value vectors and a query vector, the task is the computation of the weighted sum of the given values (output). In this setup, weights are learned by compatibility functions between key-query pairs (of dimension d_k). For a given query (Q), keys (K) and values (V) set, the *Scaled Dot-Product (SDP) Attention* function is computed as in Eq. 5.

$$Att(V, K, Q) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{5}$$

In practice, several *attentions* are computed for a given QKV set. The QKV set is first projected into h different spaces (multiple heads), where the scaled dot-product attention is computed in parallel. Resulting values for all heads are then concatenated and once again projected, yielding the layer's output. Eq. 6 and Eq. 7 illustrate the process, in which H is the set of n heads ($H = h_1, ..., h_n$) and f is a linear projection. **Self-Attention** defines the case where query and values come from same source (learning compatibility functions within the same sequence of elements).

$$MultiHead(V, K, Q) = f(Concat(H)) \tag{6}$$

$$h_i = Att(f_i(V), f_i(K), f_i(Q)) \tag{7}$$

3.3 2D-CNN: Pervasive Attention

Different from the previous models, which are based on encoder-decoder structures interfaced by attention mechanisms, this approach relies on a single 2D CNN across both sequences (no separate coding stages) [15]. Using masked convolutions, an auto-regressive model predicts the next output symbol based on a joint representation of both input and partial output sequences. Given a source-target pair $(\boldsymbol{s}, \boldsymbol{t})$ of lengths $|\boldsymbol{s}|$ and $|\boldsymbol{t}|$ respectively, tokens are first embedded in d_s and d_t dimensional spaces via look-up tables. Token embeddings $\{x_1, \ldots, x_{|\boldsymbol{s}|}\}$ and $\{y_1, \ldots, y_{|\boldsymbol{t}|}\}$ are then concatenated to form a 3D tensor $X \in R^{|\boldsymbol{t}| \times |\boldsymbol{s}| \times f_0}$, with $f_0 = d_t + d_s$, where $X_{ij} = [y_i \;\; x_j]$. Each convolutional

	#Types	#Tokens	Avg Token Length	Avg #Tokens/Sentence
FR	5,162	42,715	4.39	8.33
MB	6,633	30,556	4.18	5.96

Table 1 Statistics for the French (FR) and Mboshi (MB) parallel sentences.

layer $l \in \{1, \ldots, L\}$ of the model produces a tensor H_l of size $|\boldsymbol{t}|\times|\boldsymbol{s}|\times f_l$, where f_l is the number of output channels for that layer. To compute a distribution over the tokens in the output vocabulary, the second dimension of the tensor is used. This dimension is of variable length (given by the input sequence) and it is collapsed by max or average pooling to obtain the tensor H_L^{Pool} of size $|\boldsymbol{t}|\times f_L$. Finally, 1×1 convolution followed by a softmax operation are applied, resulting in the distribution over the target vocabulary for the next output token.

Attention mechanism: joint encoding acts as an attention-like mechanism, since individual source elements are re-encoded as the output is generated. The self-attention approach of Lin et al. [29] is applied. It computes the attention weight tensor α, of size $|\boldsymbol{t}| \times |\boldsymbol{s}|$, from the last activation tensor H_L, to pool the elements of the same tensor along the source dimension, as in Eq. 8-9, where $W_1 \in R^{f_a}$ and $W_2 \in R^{f_a \times f_L}$ are weight tensors that map the f_L dimensional features in H_L to the attention weights via an f_a dimensional intermediate representation.

$$\alpha = softmax(W_1 \tanh(H_L W_2)) \tag{8}$$

$$H_L^{\text{Att}} = \alpha H_L \tag{9}$$

4 Comparing Architectures

4.1 Experimental Settings

Data: For our experiments we use the bilingual Mboshi-French parallel corpus [19]. This is a 5,130 sentence corpus from the language documentation process of Mboshi (Bantu C25), an endangered language spoken in Congo-Brazzaville. Table 1 presents some statistics for the dataset. We use 10% of the sentences for validation, and the remaining for training.

Parameters: Across all architectures,[5] we use embeddings of size 64, batch size of 32, dropout of 0.5 and early-stopping procedure. RNN models have one layer, a bi-directional encoder, and cell size equal to the embedding size. 2D-CNN models have 3 layers, and kernel size of 3. Transformer models were extensively optimized. The presented models have 2 heads, 3 layers (encoder and decoder), warm-up of 5k steps, and use cross-entropy loss without label-smoothing. For selecting which head to use for UWS, we experimented using

[5] RNN, 2D-CNN and Transformer implementations come respectively from [7, 15, 31]. The extensive list of parameters for these architectures is available at: `https://gitlab.com/mzboito/attention_study`

different averages (between heads and layers), and selecting the head with minimum corpus ANE. While the results were not significantly different, we kept the latter approach (ANE selection).[6]

Training: For each NMT architecture, we train five models (runs) with different initialization seeds. Before segmenting, we average the produced matrices from these different runs as in Godard et al. [21]. This can be seen as an agreement over the segmentation generated by models with different weight initializations. Optimization of the models was performed in a *monolingual* condition, where a phone sequence is segmented with regards to the corresponding word sequence (transcription) in the same language (hence monolingual). This task can be seen as an automatic extraction of a pronunciation lexicon from parallel words/phones sequences. Evaluation is performed in a *bilingual* segmentation condition that corresponds to the real UWS task.

4.2 Results

Table 2 presents the scores for the UWS task[7] and ANE results. The monolingual results are shown for information only (topline). Surprisingly, RNN models outperform the more recent (2D-CNN and Transformer) approaches. One possible explanation is the lower number of parameters (in average 700k parameters are trained, while 2D-CNN needs an additional 30.79% and Transformer 5.31%). Transformer's low performance could be due to the use of several heads, which could be "distributing" alignment information across different matrices. Nonetheless, we evaluated averaged heads and single-head models, and these resulted in significant decreases in performance. This suggests that this architecture may not need to learn explicit alignment to translate, but instead it could be capturing different kinds of linguistic information. This was discussed in the original paper, and illustrated in the provided examples [36]. Also, on the decoder side, the behavior of the self-attention mechanism on phone units is unclear and under-studied so far. For the encoder, Voita et al. [37] performed after-training encoder head removal based on *head confidence*, showing that after initial training, most heads were not necessary for maintaining translation performance. Hence, we find the Multi-head mechanism interpretation challenging, and maybe not suitable for a direct UWS application, specially in low-resource settings.

Looking at the performance of these models for the monolingual scenario, which can be seen as a naive baseline in which the alignment is expected to be very *diagonal*, they all perform well. However, when the task involves discovering non-symmetrical relationships between source and target representations, the performance drops, and the attention method from Bahdanau et al. [5]

[6] We notice the same trend from Garg et al. [16], and find that the exploitable alignments are usually produced by the penultimate layer, and that these results tend to be better than simply averaging heads or layers.

[7] For 2D-CNN and RNN, average standard deviation for the bilingual task is of less than 0.8%. For Transformer, it is almost 4%.

	Bilingual				Monolingual			
	P	**R**	**F**	**ANE**	**P**	**R**	**F**	**ANE**
RNN	72.3	75.9	74.0	0.49	92.9	92.1	92.5	0.15
2D-CNN	65.9	70.6	68.2	0.64	89.6	90.1	89.8	0.19
Transformer	56.6	80.2	66.4	0.74	79.8	87.7	83.5	0.28

Table 2 UWS Boundary (Precision, Recall and F-score) and Corpus ANE scores for bilingual and monolingual settings.

ANE (<)	**Bucket 1**	**Bucket 2**	**Bucket 3**	**Bucket 4**	**All Buckets**
0.2	68.8	59.2	56.4	47.8	49.0
0.4	44.8	41.4	38.0	31.8	32.6
0.6	38.3	34.5	30.6	25.3	24.7
0.8	36.8	32.4	28.8	22.8	22.2
1	36.7	32.4	28.8	22.6	22.1

Table 3 Precision scores for Type retrieval for the alignment complexity buckets, and for the totality of the corpus (All buckets). Results in each of the rows are cumulative and use the Alignment ANE thresholds indicated in the first column.

(RNN) is the most exploitable in low-resource settings. Moreover, ANE is consistent with UWS results: ANE decreases as UWS performance increases. In Boito et al. [10], a deeper investigation of data size and language impact was performed, and a strong negative correlation was found between UWS F-scores and the corpus ANE results obtained.

To show the relation between alignment complexity and the quality of the discovered segmentation, we use `FastAlign` [14] to obtain alignment probability scores for all sentences in the Mboshi corpus, using the reference segmentation and the French text. The resulting scores can be seen as the degree of *syntactic divergence* between source and target sentences.[8] We then create four *alignment complexity buckets* of equal size for separating the corpus in four subsets with different degrees of complexity for the UWS task. For this analysis, we use the soft-alignment probability matrices produced by the RNN model. Figure 1 shows an example per bucket for sentences of equal source length: buckets one to four have increasing alignment complexity scores, with alignment probability thresholds of, respectively, -10.61, -46.87, -60.18, and -78.15.

To verify the intuition that alignment quality will deteriorate as complexity rises, we extract Alignment ANE scores for the matrices in every bucket. The alignment ANE score for a given (discovered type, translation word) pair gives us information about how confident the network is about that discovered pairing. Boito et al. [10] showed that this metric can be used for increasing Type F-scores in UWS. Here our goal is to verify the precision of type retrieval in each bucket, to check for any relation with the *straightforwardness* of the alignment task.

Table 3 shows the type retrieval precision scores for UWS using different Alignment ANE thresholds. In these results, buckets with easier examples in

[8] Results, however, are an approximation.

term of alignment probabilities (from `FastAlign`) have higher overall precision. This confirms that the quality of the alignments obtained is related to the syntactic divergence of the sentences. However, it is interesting to notice that even for the worst case (bucket 4), there are still a fair amount of high-quality alignments being retrieved. We believe this highlights the robustness of the RNNs, that even in low-resource settings, are able to learn non-trivial equivalences between source and target sentences. In the remaining of this paper, we will analyse possible methods for increasing RNN's performance for the UWS task.

5 Alignment-focused Optimization

In the last section, we investigated the impact of using different attention mechanisms for performing UWS in low-resource settings. Our experiments led us to conclude that RNNs are the most efficient for this task. In this section we focus on the investigation of methods for increasing the exploitability of the soft-alignments produced by this architecture. We investigate different training and target representation approaches for a particular low-resource scenario when 5k parallel sentences only are available.

5.1 Leveraging Monolingual Data

For language documentation scenarios, transcription is very time-consuming: one minute of audio is estimated to take one hour and a half on average of a linguist's work [4]. This is one of the motivations for the bilingual approach for UWS that we use in this paper. However, even if we cannot rely on transcriptions for data being available when treating oral-languages, it is not uncommon for a small portion of the produced documentation corpora to be manually transcribed. It might then be interesting to use this annotation, when available, as a way of *informing* the bilingual alignment process.

For this experiment, we randomly select 1,000 sentences from our corpus for which we consider we do have access to the transcription. We call it the monolingual set.[9] For incorporating the information present in this set into the training protocol, we separate it in three steps, each one trained for one third of the total number of epochs for RNNs from Section 4.1.[10]

First we train the model using only the monolingual set, making use of their gold transcriptions aligned to the unsegmented phones. This is the same scenario from the monolingual protocol in the last section, with the difference that here less data is used. Following this, the model is trained with a mixed

[9] We maintain the data protocol from the last section, keeping 10% of the sentences for validation, and the remaining for training.

[10] In our experiments we find that adopting too many epochs for the initial and intermediate steps makes the network forgetful, and the results end up being equivalent to the ones from Section 4.

representation. This mixed representation contains the monolingual set, and the 4,130 remaining sentences with bilingual alignment (no transcription). Finally, in the last step, the network is trained fully in bilingual settings (French words aligned to unsegmented phone sequences).

Lastly, for these experiments, we adapt our representation to include language tags[11] in the target side, as in Johnson et al. [27]. This is necessary because in this setup encoder annotations will vary by encoding transcriptions or bilingual text. The tags in the target side are thus a way of better informing the decoder network of the type of source annotation it will attend to. In preliminary experiments, we noticed that including language tags in the decoder increased its capacity to generate exploitable alignments.

5.2 Hybrid Bayesian-Neural Model

Non-parametric Bayesian models [24, 26] are statistical approaches that can be used for word segmentation and morphological analysis, being known as very robust in low-resource settings [18, 23]. In these monolingual models, words are generated by a uni or bigram model over a non-finite inventory, through the use of a Dirichlet process. Although providing reliable segmentation in low-resource settings, these monolingual models are incapable of automatically producing alignments with a foreign language, and therefore the discovered pseudo-word segments can be seen as "meaningless". Godard et al. [21] also showed that `dpseg`[12] [22, 23] behaves poorly on pseudo-phone units discovered from speech, which limits its application.

In this work, we investigate the use of `dpseg` as an intermediate monolingual-rooted segmentation system, whose discovered boundaries are used as clues by the bilingual neural models. This investigation derives from the notion that several intermediate segmentations might be manually produced by linguists during language documentation. We then question if the produced soft-alignments could help linguists to validate their hypotheses in this scenario.

For these experiments, we augment the original unsegmented phone sequence with the `dpseg` output boundaries. In this augmented input representation, a boundary is denoted by a special token # which separates the words identified by `dpseg`. We call this *soft-boundary insertion*, since the `dpseg` boundaries inserted into the phone sequence can be ignored by the NMT model, and new boundaries can be inserted as well. Figure 2 brings an example of soft-alignment probability matrices produced by this hybrid approach. We verify that the networks are able to ignore this symbol, keeping the alignment at the same source word, or to accept it, blurring the alignment for the # token.

[11] We use two language tags, <mono> and <bi>, for denoting unsegmented phones aligned to transcriptions and translations, respectively. These tags are added to the beginning of every sentence.

[12] Available at `http://homepages.inf.ed.ac.uk/sgwater/resources.html`

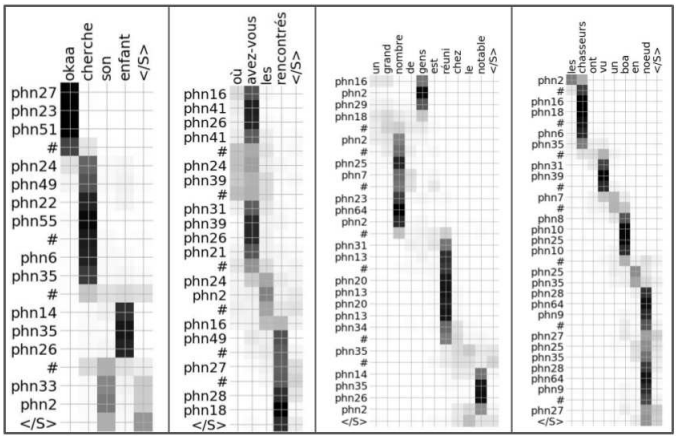

Fig. 2 Soft-alignment probability matrices for the hybrid models. The examples are ordered, from left to right, by alignment complexity buckets. The # is the soft-boundary symbol.

5.3 Word length biased NMT Training

Lastly, we investigate the RNN architecture from Godard et al. [20] that explicitly optimizes the word-length of the segmentations obtained during training. There are two differences from this model to the one presented in Section 3.1. The first one is the attention mechanism from Eq. 3, which they modify for including a bias towards longer words. They define attention over the source words as in Eq. 10, where γ is a monotonically increasing function of the source word's length given by $|w_j|$. The intuition behind this modification is that longer words should be aligned to more phone units than shorter words.

$$c_t = Att(H, s_{t-1}) = \sum_{j=1}^{|s|} \gamma(|w_j|)\alpha_{t,j}h_j \tag{10}$$

The second difference is the introduction of an auxiliary loss. The goal of this loss is to control the number of words a segmentation produces on the target side, encouraging it to become closer to the number of words in the source language. This is illustrated in Eq. 11, where $|s|$ and $|t|$ are respectively the lengths of source (word-level) and target (phone-level) sentences. The last term represents the number of target phones not segmented by the alignment generated.

$$\mathcal{L}_{AUX}(\Omega|w) = ||t| - |s| - \sum_{t=1}^{|t|-1} \alpha_{t,*}^T \alpha_{t+1,*}| \tag{11}$$

		Types	Tokens	Boundaries
1	**Base model (RNN)**	24.7	46.6	74.0
2	**dpseg (monolingual)***	*24.8*	*49.1*	*77.0*
3	**Pre-training Model**	26.2	47.8	74.8
4	**Hybrid Model**	**29.1**	**48.9**	**76.5**
5	**Word length Model**	26.6	48.1	75.2

Table 4 UWS types, tokens and boundaries F-score results for the base model (1), `dpseg` segmentation baseline (2), and optimizations (3-5). `dpseg` results are shown for reference only, since its resulting segmentation does not provide any MT alignment.

5.4 Results

Table 4 presents UWS results for the base model (RNN), the segmentation baseline (reference only), and the investigated optimizations. For the latter, rows 3 to 5 bring them in the same order they were presented (Sections 5.1 to 5.3). We do not report ANE results since the representation level for 3 and 4 are not the same for the target sequences. These methods introduce respectively language tags, and soft-boundaries into the phone sequences. This makes ANE scores not comparable.

Looking at the results, we notice that the hybrid optimization achieves the best segmentation results, followed by the word length optimization and the pre-training model. We remind the reader that here the UWS task is an extrinsic metric for assessing alignment quality for NMT models. The `dpseg` baseline reaches higher token and boundary F-scores, compared to the hybrid model, but its segmentation does not come from the training of a translation system. Therefore results are presented only to assess the quality of the information injected into the input representation of the hybrid model.

Focusing at the models investigated in this section, we notice that the pre-training model is the worst among them. We analysed its type retrieval scores, to verify if the network successfully *remembered* the types for which the reference segmentation was provided. We found that, from the first to the last step, the retrieval for these types dropped 23.6% (from 56.2% to 32.6%). This hints that, even if some information was propagated to the final bilingual model, this method could still benefit from a more direct method of segmentation *bias*. An alternative is the work performed in Boito et al. [9], in which the pre-segmentation for the 100 most frequent types was performed before NMT training.[13] However, we believe the latter to be sub-optimal, since sub-word information is potentially lost. Also, in this setting encoder networks must deal with a mixed representation of words and phones. The flexibility of not forcing a segmentation, and yet informing the model about possible boundaries, might be the reason why the hybrid model performs the best.

Still about this hybrid model, we turn our attention to the examples in Figure 2. There, we can observe that the existence of boundary clues adds some disturbance to the produced matrices. This is noticeable by the brighter

[13] This comes from the intuition that, in documentation scenarios, this would reflect the knowledge acquired by a linguist after some days in the community.

ANE (<)	Bucket 1	Bucket 2	Bucket 3	Bucket 4	All buckets
0.2	82.0 (+13.3)	66.7 (+7.4)	69.0 (+12.5)	73.3 (+25.5)	64.5 (+15.5)
0.4	59.3 (+14.5)	55.4 (+14.1)	51.3 (+13.2)	45.6 (+13.8)	45.9 (+13.2)
0.6	47.9 (+9.6)	44.6 (+10.2)	40.8 (+10.2)	33.7 (+8.4)	32.7(+8.0)
0.8	43.9 (+7.1)	40.7 (+8.2)	37.1 (+8.3)	29.2 (+6.4)	28.2 (+6.0)
1	43.7 (+7.0)	40.2 (+7.8)	36.9 (+8.1)	28.9 (+6.3)	28.0 (+6.0)

Table 5 Precision scores in type retrieval for the alignment complexity buckets, and for the totality of the corpus (All buckets) using the matrices produced by the hybrid model. Results are cumulative and use the Alignment ANE thresholds indicated in the first column. The absolute difference between the obtained scores and the ones from the base model (Table 3) is displayed between parenthesis.

square colors for the #'s probability distributions. Even so, the network seems capable of ignoring these clues when necessary. We perform the same analysis from Section 4, investigating precision scores in type retrieval for the alignment complexity buckets. The results, presented in Table 5, show an expressive difference in type precision, compared to the base model. The augmented input representation seems to have helped this model especially in more challenging alignment scenarios (bucket 3 and 4). The hybrid model also increased type F-score over the segmentation baseline `dpseg`. This suggests that the boundary clues *informed* the network, instead of just forcing a pre-established segmentation, which resulted in more meaningful source-to-target alignments.

Finally, for the word length optimization method, it may have suffered from over-constraining the produced alignment. This method forces the amount of words produced to be close to the number of source words available, what ends up reducing the flexibility of the attention mechanism. For instance, in the third example in Figure 2, we see that some source words are almost completely ignored. This may need to happen when source and target languages differ syntactically. Moreover, in the case of non-existing translation for a given word, linguists might translate it by giving an *explanation* to the term. This would result in a case of many-to-one alignment that would result in over-segmentation.

Summarizing, we investigated different methods for increasing the quality and usability of the soft-alignment probability matrices discovered by NMT RNN-based models. We find that adding boundary clues to the input representation is the best way of informing the neural model, resulting in the best UWS results. This suggests that, in a documentation scenario, `dpseg` could be replaced by early annotations of potential words done by a linguist, for instance. The linguist could then validate the output of the neural system, and review their word hypotheses considering the generated bilingual alignment.

6 Conclusion

In this paper we investigated the interpretability of attention-based NMT architectures for low-resource settings. Our focus lies on the direct exploitation of source-to-target alignment, evaluating the soft-alignment probability matrices

produced by NMT models with respect to their performance in the Unsupervised Word Segmentation (UWS) task. For this task, words are translated into unsegmented phone sequences, and the alignments obtained by the NMT training must result in segmentation over the target side. We compared three well-known attention-based NMT models (RNN, 2D-CNN and Transformer), finding that the RNN achieved the best results in low-resource settings. We also evaluated the Average Normalized Entropy (ANE) for the ensemble of soft-alignment matrices produced by the different models. The results obtained highlight the correlation between low ANE scores and higher segmentation scores, and better alignment quality.

We followed this by investigating methods for increasing the exploitability of the soft-alignment probability matrices produced by the RNN architecture. We investigated the following methods: pre-training, a hybrid approach which includes soft-boundaries in the input representation, and a word length alignment optimization during training. Interestingly, we found the hybrid approach to be the most efficient. This approach was superior in type retrieval to the strong segmentation baseline `dpseg`. We attribute its efficiency to the flexibility it allows, since the NMT models are not forced to respect a given segmentation, and instead the boundary clues are used as bias. We hypothesise that boundary information, even when noisy, can help the internal representation of the RNN models. In documentation scenarios, this supervision could come from linguists, during the documentation process. Lastly, in line with the results by Boito et al. [10], this work also confirms that ANE can be used as a threshold for extracting *high-confidence* alignments, which can help linguists to filter the generated bilingual vocabulary. The insertion of boundary clues is yet another way to collaborate with linguists during this process.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Adda, G., Stüker, S., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.N., Lamel, L., Makasso, E.M., Rialland, A., de Velde, M.V., Yvon, F., Zerbian, S.: Breaking the unwritten language barrier: The BULB project. Procedia Computer Science **81**, 8–14 (2016)
2. Anastasopoulos, A., Chiang, D.: A case study on using speech-to-translation alignments for language documentation. arXiv preprint arXiv:1702.04372 (2017)
3. Austin, P.K., Sallabank, J.: The Cambridge handbook of endangered languages. Cambridge University Press (2011)
4. Austin, P.K., Sallabank, J.: Endangered languages. Taylor & Francis (2013)
5. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
6. Bartels, C., Wang, W., Mitra, V., Richey, C., Kathol, A., Vergyri, D., Bratt, H., Hung, C.: Toward human-assisted lexical unit discovery without text resources. In: Spoken Language Technology Workshop (SLT), 2016 IEEE, pp. 64–70. IEEE (2016)

7. Bérard, A., Pietquin, O., Servan, C., Besacier, L.: Listen and translate: A proof of concept for end-to-end speech-to-text translation. arXiv preprint arXiv:1612.01744 (2016)
8. Besacier, L., Zhou, B., Gao, Y.: Towards speech translation of non written languages. In: Spoken Language Technology Workshop, 2006. IEEE, pp. 222–225. IEEE (2006)
9. Boito, M.Z., Bérard, A., Villavicencio, A., Besacier, L.: Unwritten languages demand attention too! word discovery with encoder-decoder models. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp. 458–465. IEEE (2017)
10. Boito, M.Z., Villavicencio, A., Besacier, L.: Empirical evaluation of sequence-to-sequence models for word discovery in low-resource settings. arXiv preprint arXiv:1907.00184 (2019)
11. Chorowski, J.K., Bahdanau, D., Serdyuk, D., Cho, K., Bengio, Y.: Attention-based models for speech recognition. In: Advances in neural information processing systems, pp. 577–585 (2015)
12. Dunbar, E., Cao, X.N., Benjumea, J., Karadayi, J., Bernard, M., Besacier, L., Anguera, X., Dupoux, E.: The zero resource speech challenge 2017. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp. 323–330. IEEE (2017)
13. Duong, L., Anastasopoulos, A., Chiang, D., Bird, S., Cohn, T.: An attentional model for speech translation without transcription. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 949–959 (2016)
14. Dyer, C., Chahuneau, V., Smith, N.A.: A simple, fast, and effective reparameterization of ibm model 2. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 644–648 (2013)
15. Elbayad, M., Besacier, L., Verbeek, J.: Pervasive attention: 2d convolutional neural networks for sequence-to-sequence prediction. arXiv preprint arXiv:1808.03867 (2018)
16. Garg, S., Peitz, S., Nallasamy, U., Paulik, M.: Jointly learning to align and translate with transformer models. arXiv preprint arXiv:1909.02074 (2019)
17. Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N.: Convolutional sequence to sequence learning. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 1243–1252. JMLR. org (2017)
18. Godard, P., Adda, G., Adda-Decker, M., Allauzen, A., Besacier, L., Bonneau-Maynard, H., Kouarata, G.N., Löser, K., Rialland, A., Yvon, F.: Preliminary experiments on unsupervised word discovery in mboshi. In: Proc. Interspeech (2016)
19. Godard, P., Adda, G., Adda-Decker, M., Benjumea, J., Besacier, L., Cooper-Leavitt, J., Kouarata, G.N., Lamel, L., Maynard, H., Müller, M., et al.: A very low resource language speech corpus for computational language documentation experiments. arXiv preprint arXiv:1710.03501 (2017)
20. Godard, P., Besacier, L., Yvon, F.: Controlling utterance length in nmt-based word segmentation with attention. arXiv preprint arXiv:1910.08418 (2019)
21. Godard, P., Zanon Boito, M., Ondel, L., Berard, A., Yvon, F., Villavicencio, A., Besacier, L.: Unsupervised word segmentation from speech with attention. In: Interspeech (2018)
22. Goldwater, S., Griffiths, T.L., Johnson, M.: Contextual dependencies in unsupervised word segmentation. In: Proc. International Conference on Computational Linguistics, pp. 673–680. Sydney, Australia (2006)
23. Goldwater, S., Griffiths, T.L., Johnson, M.: A Bayesian framework for word segmentation: Exploring the effects of context. Cognition **112**(1), 21–54 (2009)
24. Goldwater, S.J.: Nonparametric bayesian models of lexical acquisition. Ph.D. thesis, Citeseer (2007)
25. Jain, S., Wallace, B.C.: Attention is not explanation. arXiv preprint arXiv:1902.10186 (2019)
26. Johnson, M., Goldwater, S.: Improving nonparameteric bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In: Proc. NAACL-HLT, pp. 317–325. Association for Computational Linguistics (2009)
27. Johnson, M., Schuster, M., Le, Q.V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al.: Google's multilingual neural machine translation system: Enabling zero-shot translation. Transactions of the Association for Computational Linguistics **5**, 339–351 (2017)

28. Lignos, C., Yang, C.: Recession segmentation: simpler online word segmentation using limited resources. In: Proceedings of the fourteenth conference on computational natural language learning, pp. 88–97. Association for Computational Linguistics (2010)
29. Lin, Z., Feng, M., dos Santos, C., Yu, M., Xiang, B., Zhou, B., Bengio, Y.: A structured self-attentive sentence embedding. In: ICLR (2017)
30. Ondel, L., Burget, L., Černockỳ, J.: Variational inference for acoustic unit discovery. Procedia Computer Science **81**, 80–86 (2016)
31. Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., Auli, M.: fairseq: A fast, extensible toolkit for sequence modeling. arXiv preprint arXiv:1904.01038 (2019)
32. Serrano, S., Smith, N.A.: Is attention interpretable? arXiv preprint arXiv:1906.03731 (2019)
33. Shen, J., Pang, R., Weiss, R.J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., et al.: Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4779–4783. IEEE (2018)
34. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, K.Q. Weinberger (eds.) Advances in Neural Information Processing Systems 27, pp. 3104–3112. Curran Associates, Inc. (2014). URL `http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf`
35. Vashishth, S., Upadhyay, S., Tomar, G.S., Faruqui, M.: Attention interpretability across nlp tasks. arXiv preprint arXiv:1909.11218 (2019)
36. 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)
37. Voita, E., Talbot, D., Moiseev, F., Sennrich, R., Titov, I.: Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. arXiv preprint arXiv:1905.09418 (2019)
38. Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R.J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., et al.: Tacotron: Towards end-to-end speech synthesis. arXiv preprint arXiv:1703.10135 (2017)
39. Watanabe, S., Hori, T., Kim, S., Hershey, J.R., Hayashi, T.: Hybrid ctc/attention architecture for end-to-end speech recognition. IEEE Journal of Selected Topics in Signal Processing **11**(8), 1240–1253 (2017)
40. Wiegreffe, S., Pinter, Y.: Attention is not not explanation. arXiv preprint arXiv:1908.04626 (2019)
41. Zenkel, T., Wuebker, J., DeNero, J.: Adding interpretable attention to neural translation models improves word alignment. arXiv preprint arXiv:1901.11359 (2019)