# ON-TRAC Consortium End-to-End Speech Translation Systems for the IWSLT 2019 Shared Task

*Ha Nguyen*[1], *Natalia Tomashenko*[2], *Marcely Zanon Boito*[1], *Antoine Caubrière*[3], *Fethi Bougares*[3], *Mickael Rouvier*[2], *Laurent Besacier*[1], *Yannick Estève*[2]

[1]LIG - Université Grenoble Alpes, France
[2]LIA - Avignon Université, France
[3]LIUM - Le Mans Université, France

## Abstract

This paper describes the ON-TRAC Consortium translation systems developed for the end-to-end model task of IWSLT Evaluation 2019 for the English→ Portuguese language pair. ON-TRAC Consortium is composed of researchers from three French academic laboratories: LIA (Avignon Université), LIG (Université Grenoble Alpes), and LIUM (Le Mans Université). A single end-to-end model built as a neural encoder-decoder architecture with attention mechanism was used for two primary submissions corresponding to the two EN-PT evaluations sets: (1) TED (MuST-C) and (2) How2. In this paper, we notably investigate impact of pooling heterogeneous corpora for training, impact of target tokenization (characters or BPEs), impact of speech input segmentation and we also compare our best end-to-end model (BLEU of 26.91 on MuST-C and 43.82 on How2 validation sets) to a pipeline (ASR+MT) approach.

## 1. Introduction

Previous automatic speech-to-text translation (AST) systems operate in two steps: source language speech recognition (ASR) and source-to-target text translation (MT). However, recent works have attempted to build end-to-end AST without using source language transcription during learning or decoding [1, 2] or using it at training time only [3]. Very recently several extensions of these pioneering works were introduced: low-resource AST [4], unsupervised AST [5], end-to-end speech-to-speech translation (*Translatotron*) [6]. Improvements of end-to-end AST were also proposed using weakly supervised data [7], or by adding a second attention mechanism [8].

This paper describes the ON-TRAC consortium automatic speech translation (AST) systems for the IWSLT 2019 Shared Task. ON-TRAC Consortium is composed of researchers from three French academic laboratories: LIA (Avignon Université), LIG (Université Grenoble Alpes), and LIUM (Le Mans Université).

We participated to the end-to-end model English-to-Portuguese AST task on *How2* [9] and *MuST-C* [10] datasets. We notably try to answer to the following questions:

- Question 1: does pooling heterogenous corpora (*How2* and *MuST-C*) help the AST training?
- Question 2: what is the better tokenization unit on the target side (BPE or characters)?
- Question 3: considering that segmentation is an important challenge of AST, what is the optimal way to segment the speech input?
- Question 4: does fine-tuning increase the system's performance?
- Question 5: is our end-to-end AST model better than an ASR+MT pipeline?

This paper is organized as follows: after briefly presenting the data in Section 2 and after detailing our investigation on automatic speech segmentation in Section 3, we present the end-to-end speech translation systems submitted by our ON-TRAC consortium in Section 4. Section 5 summarizes what we learned from this evaluation and Section 6 concludes this work.

## 2. Data

The corpora used in this work are the How2 [9] and MuST-C [10] corpora. Since we focus on English-to-Portuguese AST tasks, only the English-Portuguese portion of MuST-C corpus is used. The statistics of these two corpora, along with the corresponding provided evaluation data, can be found in Table 1. In order to answer to the first scientific question mentioned in Section 1, we pool these two corpora together to create a merged corpus whose details can also be found in the same table.

Note that the statistics for the How2 training set might slightly differ from that of other participants since the original audio files for the How2 corpus are not officially available. Since we wanted to apply our own feature extraction, instead of using the one shared by the How2 authors, we have to download the original video files from Youtube,[1] and then

[1]https://www.youtube.com/

| Corpus | #Segments | Hours | #src words | #tgt words |
|---|---|---|---|---|
| MuST-C | 206,155 | 376.8 | 3.9M | 3.7M |
| How2 | 184,624 | 297.6 | 3.3M | 3.1M |
| Merged corpus | 390,779 | 674.4 | 7.2M | 6.8M |
| MuST-C eval | 2,571 | 5.4 | - | - |
| How2 eval | 2,497 | 4.5 | - | - |

Table 1: Statistics of the original MuST-C and How2 corpora, the merged version, and the official evaluation data (audio data only).

extract the audio from these downloaded video files. One issue with this approach is that the final corpus content will depend on the availability of audio files on Youtube at the downloading date. On July 12th, when our version of the corpus was downloaded, 21 (out of 13,472) video files were missing. We consider this as a minor loss with regard to the possibility it gives us to extract our own acoustic features.

## 3. Speech segmentation

While How2 evaluation data is distributed with a predefined segmentation, this information is not provided for the TED talks evaluation data. In this context, we explore two different approaches to segment the MuST-C (TED talks) audio stream. The first one is based on the use of the well known LIUM_SpkDiarization toolkit [11], which is an open source toolkit for speaker diarization (we used the default configuration).

The second approach is based on the use of an Automatic Speech Recognition system (ASR) as a speech segmenter: we transcribe automatically and without segmentation all the validation and evaluation datasets with a Kaldi-based ASR system [12] trained on TEDLIUM 3 [13].[2] We did not try to optimize the ASR system on our data. This ASR system produces recognized words with timecodes (start time and duration for each word). Thanks to this temporal information, we are able to measure silence duration between two words when silence (or non speech event) exists. When a silence between two words is higher than 0.65 seconds, we split the audio file. When the number of words in the current speech segment exceeds 40, this threshold is reduced to 0.15 seconds, in order to avoid exceedingly long segments. These thresholds have been tuned in order to get a segment duration distribution in the evaluation data close to the one observed in the training data. Table 2 summarizes statistics about segment duration on training data (with the segmentation provided by the organizers) and evaluation data (ASR-based segmentation *vs.* speaker diarization toolkit).

In order to choose the segmentation process for our primary system among these two approaches, we carried out

[2]In the context of the campaign, the use of some of TEDLIUM 3 files is forbidden. These files have been removed before training the ASR system.

| Corpus/Segmentation | min size | max size | average | std dev |
|---|---|---|---|---|
| Train/Organizers | 0.17 | 30.00 | 6.31 | 4.72 |
| Eval/ASR-based | 0.03 | 22.71 | 6.09 | 4.52 |
| Eval/SpkDiar | 1.51 | 20.00 | 9.62 | 5.33 |

Table 2: Statistics on speech segments duration (MuST-C) for 2 different segmentation approaches. All values are given in seconds.

experiments on the tst-COMMON data from the MuST-C corpus. For these experiments, we applied a preliminary version of our end-to-end system, trained on the MuST-C training data to translate speech into lower-case text. Then, we used the *mwerSegmenter* tool[3] to realign our translations to the reference segmentation of the tst-COMMON data, in order to evaluate translation quality. Table 3 shows the BLEU score obtained with different segmentation strategies: manual (original MuST-C annotations), ASR-based, and speaker diarization.

| Segmentation | BLEU |
|---|---|
| Manual (original) | 25.50 |
| Speaker Diarization [11] | 21.01 |
| ASR-based | 22.03 |

Table 3: BLEU scores (lower-case evaluation) obtained on the tst-COMMON (MuST-C corpus) data with different speech segmentation strategies.

Those preliminary results show that ASR-based segmentation leads to better speech translation performance than the speaker diarization approach. However, we observe that manual segmentation (25.50) still outperforms our best automatic segmentation (22.03). This shows that automatic segmentation of the audio stream is an important issue to address for the speech translation task.

Finally, based on these findings we decided to use the ASR-based approach for our primary system applied to the TED talks (MuST-C) evaluation data (for which we do not possess manual segmentation). For the How2 evaluation data, we use the manual segmentation provided by the organizers.

## 4. Speech translation systems

In this work, several speech translation systems were developed for translating English speech into Portuguese text (EN-PT).

### 4.1. End-to-end speech translation

In this section we detail our end-to-end architecture. All the experiments presented are conducted using the ESPnet [14]

[3]`https://www-i6.informatik.rwth-aachen.de/web/Software/mwerSegmenter.tar.gz`

end-to-end speech processing toolkit.

**Speech features.** For all models, 80-dimensional Mel filter-bank features, concatenated with 3-dimensional pitch features,[4] are used for training. Features are extracted using $25ms$ windows with a frame shift of $10ms$. Cepstral mean and variance normalization is computed on the training set. Data augmentation, based on speed perturbation with factors of 0.9, 1.0, and 1.1, is applied to the training data [16].

**Text preprocessing.** Following the ESPnet speech translation recipe, we normalize punctuation, and tokenize all the Portuguese text using Moses.[5] Texts are case-sensitive and contain punctuation. Moreover, the texts of the MuST-C corpus contain *'Laughter'*, *'Applause'* marks. These are kept for training the model which uses only MuST-C data, but they are removed from the texts when training the models on the combination of both corpora to ensure consistency.

The development sets are generated by randomly sampling 2,000, 2,000, and 4,000 sentences from MuST-C, How2 and the merged corpus respectively. These sentences are removed from the corresponding training sets.

Furthermore, to make the training feasible with our limited computational resources, training and development sentences longer than $3,000$ frames ($\approx 30s$) or $400$ characters are removed. This results in $6\%$, $8\%$ and $7\%$ speech data loss for How2, MuST-C and the merged corpus respectively.

The summarization of the training data after preprocessing can be found in Table 4.

| Set | #Segments | #src words | #tgt words |
|---|---|---|---|
| MuST-C train | 597,871 | 10.9M | 10.3M |
| MuST-C dev | 1,994 | 36.4K | 34.4K |
| How2 train | 538,231 | 9.4M | 8.9M |
| How2 dev | 1,984 | 33.7K | 32.0K |
| Merged train | 1,136,084 | 20.9M | 19.2M |
| Merged dev | 3,978 | 72.4K | 66.5K |

Table 4: Statistics for the training data after preprocessing.

**Architecture.** We use an attention-based encoder-decoder architecture, whose encoder has two VGG-like [17] CNN blocks followed by five stacked 1024-dimensional BLSTM layers (see Figure 1). The decoder has two 1024-dimensional LSTM layers. Each VGG block contains two 2D-convolution layers followed by a 2D-maxpooling layer whose aim is to reduce both time ($T$) and frequency dimension ($D$) of the input speech features by a factor of 2. These two VGG blocks transform input speech features' shape from $(T \times D)$ to $(T/4 \times D/4)$. Bahdanau's attention mechanism [18] is used in all our experiments.

[4]Pitch-features are computed using the Kaldi toolkit [12] and consist of the following values [15]: (1) probability of voicing (POV-feature), (2) pitch-feature and (3) delta-pitch feature. For details, see `http://kaldi-asr.org/doc/process-kaldi-pitch-feats_8cc.html`

[5]`http://www.statmt.org/moses/`

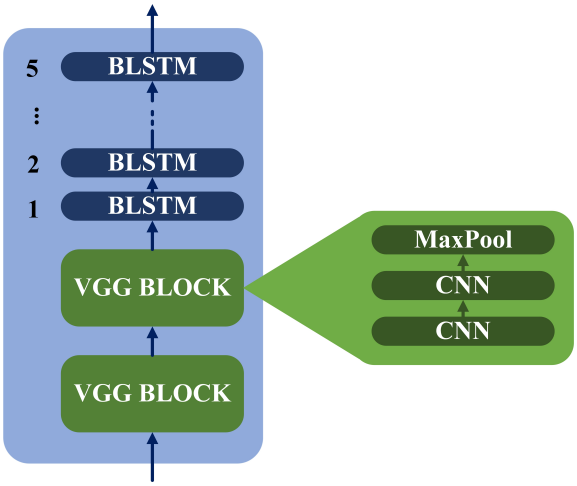


Figure 1: Architecture of the speech encoder: a stack of two VGG blocks followed by 5 BLSTM layers.

**Hyperparameters' details.** In all our experiments, dropout is set only on the encoder part with the probability of 0.3. Adadelta is chosen as our optimizer. All end-to-end models developed in this paper have similar architectures and differ mainly in the following aspects: (1) training corpus; (2) type of tokenization units; (3) fine-tuning and pretraining strategies. Description of different models and evaluation results are given in Section 5.

### 4.2. Pipeline approach (baseline)

In this section we describe the pipeline approach for speech translation.

**ASR system.** Kaldi speech recognition tookit [19] was used for this purpose. The system used in the pipeline is close to the *tedlium/s5_r3* recipe.[6] The acoustic model is trained on TEDLIUM-3 and a subset of MuST-C corpus. We use TDNN-F (11 TDNN-F layers) structures for acoustic modeling with 40-dimensional MFCC features. A simple 3-gram language model (LM) is trained using TEDLIUM-3, MuST-C and How2 corpus, with SRILM toolkit [20]. The ASR system achieved a case-insensitive Word Error Rate (WER) of 21.71% and 26.89% on *Must-C tst-COMMON* and *How2 val* sets respectively.

**MT system.** we used the Transformer [21] sequence-to-sequence model as implemented in `fairseq` [22]. Transformer is the state of the art NMT model. In this architecture, scaled-dot-product attention between keys, values and query vectors in multiple dimensions (or heads) is computed. This is done both *within* encoder and decoder stacks (multi-head self attention) and *between* encoder and decoder stacks (multi-head encoder-decoder attention).

Our models are based on the small transformer settings using 6 stacks (layers) for encoder and decoder networks

[6]`https://github.com/kaldi-asr/kaldi/tree/master/egs/tedlium/s5_r3/`

with an embedding layer of size 512, a feed-forward layer with an inner dimension of 1024, and 4 heads for the multi-head attention layers. We train the NMT system using the merged corpora (Table 1) with a vocabulary of 30K units based on a joint source and target byte pair encoding (BPE) [23]. Results of the pipeline speech translation system are reported in the last line of Table 5.

| Evaluation set | ASR | Ref |
|---|---|---|
| How2 val | 34.23 | 51.37 |
| MuST-C tst-COMMON | 22.14 | 28.34 |

Table 5: Detokenized Case-sensitive BLEU scores for different evaluation sets when translating the automatic (ASR) and human (Ref) transcription.

## 5. Experiments and lessons learned

In order to answer the scientific questions introduced in Section 1, we conducted a series of experiments whose results are presented in Table 6.

### 5.1. Question 1: choosing the training corpus

We train three end-to-end models with the architecture described in Section 4 using three different training corpora: (1) MuST-C, (2) How2, and (3) the merged version of the two corpora. The target tokens are characters. These models are then evaluated on the *tst-COMMON* (MuST-C), and *val* (How2) datasets, and the results are reported in the first three lines of Table 6. We can observe that the model trained on the merged corpora outperforms the ones trained on MuST-C (difference of 3.32) and How2 (difference of 3.11). This model (line #3 of the table) is used for our IWSLT primary system submission for both evaluation datasets.

### 5.2. Question 2: choosing the tokenization units

In this series of experiments, we investigate the impact of the tokenization units on the performance of the translation system. We investigated two types of tokenization units: characters and subword units based on byte-pair encoding (BPE) [23]. Using BPE units, we train four models with different vocabulary sizes: 400, 2,000, 5,000 and 8,000. Results for the models are given in Table 6, lines #4–7, in which we observe that having fewer output tokens on the decoder side is beneficial. We conclude that characters seem to be the best tokenization units on the MuST-C, and BPE 400 units provides the best results for the How2 task.[7]

### 5.3. Question 3: segmentation

We have seen in Section 3 that our ASR-based segmentation leads to better BLEU scores than using off-the-shelf speaker diarization. Our primary system used the ASR-based segmentation to process TED talks, while a contrastive system used speaker diarization. We expect that the final campaign results will confirm our preliminary conclusion.[8]

[7]However, since the bpe-400 result for How2 was obtained after the evaluation deadline, our official submission uses characters for both datasets).

| No. | Experiment | Token | MuST-C tst-COMMON | How2 val |
|---|---|---|---|---|
| 1 | Must-C | char | 23.59 | - |
| 2 | How2 | char | - | 39.86 |
| 3* | Merged | char | **26.91** | 42.97 |
| 4 | Merged | bpe-400 | 24.73 | **43.82** |
| 5 | Merged | bpe-2k | 23.11 | 41.45 |
| 6 | Merged | bpe-5k | 22.25 | 41.20 |
| 7 | Merged | bpe-8k | 21.75 | 40.07 |
| 8 | FT / Unfreeze | char | - | 43.02 |
| 9 | FT / Freeze | char | - | 43.04 |
| 10 | Pipeline (table 5) | bpe-30k | 22.14 | 34.23 |

Table 6: Detokenized case-sensitive BLEU scores for different experiments. Two lines with $FT$ correspond to the models trained on the merged training corpus and fine-tuned (FT) using only the How2 corpus.

### 5.4. Question 4: fine-tuning impact

We also investigate fine-tuning. For instance, training for one more epoch on the target corpus might help to improve translation performance. In order to verify this, we extend the training of the model which uses the merged corpora (line #3 in Table 6) for one more epoch on the How2 corpus only (our evaluation target). We investigated (1) fine-tuning both encoder and decoder (*Unfreeze*, line #8) and (2) fine-tuning the decoder only (*Freeze*, line #9). Results are presented at the last two lines of Table 6. We observe a slight but not significant gain with fine-tuning and no difference between *Freeze* and *Unfreeze* options.

### 5.5. Question 5: pipeline or end-to-end

The pipeline results for both corpora are available in the last line (#10) of Table 6. We verify that our best end-to-end speech translation results (lines #3 and #4) outperform this baseline model by a difference of 4.77 points for TED talks and 9.59 points for How2. While it is important to mention that we did not fully optimize ASR, NMT systems and their combination,[9] we find that these results highlight the performance of our end-to-end speech translation systems.

## 6. Conclusion

This paper described the ON-TRAC consortium submission to the end-to-end speech translation systems for the IWSLT 2019 shared task. Our primary end-to-end translation

[8]This was written before the evaluation campaign final results release.

[9]ASR and MT were developed independently of each other in two different research groups

model used in the IWSLT-2019, evaluated on the development datasets, scores the following results for case-sensitive BLEU score: 26.91 on TED talks task and 43.02 on How2. For the How2 task, we verified (after the evaluation campaign deadline) that it is possible to obtain a better result by using the model with 400 BPE units.

## 7. Acknowledgements

This work was funded by the French Research Agency (ANR) through the ON-TRAC project under contract number ANR-18-CE23-0021.

## 8. References

[1] A. Berard, O. Pietquin, C. Servan, and L. Besacier, "Listen and translate: A proof of concept for end-to-end speech-to-text translation," in *NIPS Workshop on End-to-end Learning for Speech and Audio Processing*, Barcelona, Spain, December 2016.

[2] R. J. Weiss, J. Chorowski, N. Jaitly, Y. Wu, and Z. Chen, "Sequence-to-sequence models can directly transcribe foreign speech," *arXiv preprint arXiv:1703.08581*, 2017.

[3] A. Bérard, L. Besacier, A. C. Kocabiyikoglu, and O. Pietquin, "End-to-End Automatic Speech Translation of Audiobooks," in *ICASSP 2018 - IEEE International Conference on Acoustics, Speech and Signal Processing*, Calgary, Alberta, Canada, Apr. 2018. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01709586

[4] S. Bansal, H. Kamper, K. Livescu, A. Lopez, and S. Goldwater, "Pre-training on high-resource speech recognition improves low-resource speech-to-text translation," *CoRR*, vol. abs/1809.01431, 2018. [Online]. Available: http://arxiv.org/abs/1809.01431

[5] Y. Chung, W. Weng, S. Tong, and J. Glass, "Towards unsupervised speech-to-text translation," *CoRR*, vol. abs/1811.01307, 2018. [Online]. Available: http://arxiv.org/abs/1811.01307

[6] Y. Jia, R. J. Weiss, F. Biadsy, W. Macherey, M. Johnson, Z. Chen, and Y. Wu, "Direct speech-to-speech translation with a sequence-to-sequence model," *CoRR*, vol. abs/1904.06037, 2019. [Online]. Available: http://arxiv.org/abs/1904.06037

[7] Y. Jia, M. Johnson, W. Macherey, R. J. Weiss, Y. Cao, C. Chiu, N. Ari, S. Laurenzo, and Y. Wu, "Leveraging weakly supervised data to improve end-to-end speech-to-text translation," *CoRR*, vol. abs/1811.02050, 2018. [Online]. Available: http://arxiv.org/abs/1811.02050

[8] M. Sperber, G. Neubig, J. Niehues, and A. Waibel, "Attention-passing models for robust and data-efficient end-to-end speech translation," *CoRR*, vol. abs/1904.07209, 2019. [Online]. Available: http://arxiv.org/abs/1904.07209

[9] R. Sanabria, O. Caglayan, S. Palaskar, D. Elliott, L. Barrault, L. Specia, and F. Metze, "How2: a large-scale dataset for multimodal language understanding," *arXiv preprint arXiv:1811.00347*, 2018.

[10] M. A. Di Gangi, R. Cattoni, L. Bentivogli, M. Negri, and M. Turchi, "Must-c: a multilingual speech translation corpus," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 2012–2017.

[11] S. Meignier and T. Merlin, "Lium spkdiarization: An open source toolkit for diarization," in *CMU SPUD Workshop*, 2010.

[12] P. Daniel, G. Arnab, B. Gilles, B. Lukas, and G. Ondrej, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584, 2011.

[13] F. Hernandez, V. Nguyen, S. Ghannay, N. Tomashenko, and Y. Estève, "Ted-lium 3: twice as much data and corpus repartition for experiments on speaker adaptation," in *International Conference on Speech and Computer*. Springer, 2018, pp. 198–208.

[14] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, *et al.*, "Espnet: End-to-end speech processing toolkit," *arXiv preprint arXiv:1804.00015*, 2018.

[15] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 2494–2498.

[16] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[18] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," in *ICLR 2015*, San Diego, California, USA, 2015, pp. 3104–3112.

[19] D. Povey, A. Ghoshal, G. Boulianne, N. Goel, M. Hannemann, Y. Qian, P. Schwarz, and G. Stemmer, "The kaldi speech recognition toolkit," in *In IEEE 2011 workshop*, 2011.

[20] A. Stolcke, "Srilm – an extensible language modeling toolkit," in *IN PROCEEDINGS OF THE 7TH INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING (ICSLP 2002*, 2002, pp. 901–904.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 5998–6008. [Online]. Available: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf

[22] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," in *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[23] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1715–1725.