Automatic Speech Translation for Multinational First Responder Teams

Guillermo Cámbara*
NLP Group
Pompeu Fabra University, Barcelona
guillermo.cambara@upf.edu

Jens Grivolla
NLP Group
Pompeu Fabra University, Barcelona
jens.grivolla@upf.edu

Mireia Farrús
CLiC
University of Barcelona
mfarrus@ub.edu

Leo Wanner
Catalan Institute for Research and Advanced Studies (ICREA) and NLP Group, Pompeu Fabra University, Barcelona
leo.wanner@upf.edu

ABSTRACT

Big disasters as increasingly observed all over the world, often require the involvement of a large number of personnel, in particular personnel acting in the field, i.e., First Responders. By far not always local teams are sufficient. As a consequence, in particular in Europe, teams from different member states are dispatched to support the local teams. However, this bears a potential of miscommunication since it cannot be taken for granted that English is mastered to a sufficient degree by everybody to serve as *lingua franca*. In this paper, we present work in progress carried out in the context of the INGENIOUS project on an automatic speech translation module that facilitates the interaction between First Responders speaking different languages. The module is embedded into the Telegram Messenger Application and consists of three main modules: Automatic Speech Recognition, Machine Translation, and Text-to-Speech, which are applied in sequence. We opt for a pipeline solution instead of end-to-end speech translation in order to guarantee the availability of the original speech transcriptions and their translations.

Keywords

automatic speech translation, first responders, disaster management.

INTRODUCTION

Big disasters, as increasingly witnessed all over the world,[1] require a large number of First Responder (FR) personnel in the field. By far not always local teams suffice. Therefore, especially in Europe, it is rather common that several countries deploy personnel to support local teams. For instance, 21 EU Member States and 3 UCPM[2] Participating States deployed FR teams to the recent earthquake region in Turkey; in the summer of 2022, teams from six European countries actively participated in the extinction of the wildfire in the Dadia National Park and on the island of Lesbos in Greece; and so on. As a rule, the members of the different teams communicate with each other in English. Only that by far not always a sufficiently high English language proficiency can be guaranteed, such that understanding problems or misinterpretations may jeopardize an efficient completion of the mission or even put both the members of the FR teams and citizens in danger. With the advances in Natural Language Processing (NLP) technologies, this communication bottleneck can be addressed by developing an automatic translation service of the speech in language $\mathcal{L}_1$, spoken by a member of team $\mathcal{T}_1$, into $\mathcal{L}_2$, spoken by a member of team $\mathcal{T}_2$.

In the past, several works already addressed the problem of automatic speech translation in emergency contexts. For instance, *NineOneOne* (Nallasamy, Black, Schultz, and R. E. Frederking 2008) and *Ayudame* (Nallasamy, Black,

*corresponding author

[1] Recall the very recent earthquake in Turkey and Syria and the wildfire in Chile, the wildfires in Greece in 2022 and 2018 and in Australia in 2020 and 2019, the floods in Germany, Pakistan and other countries in 2022 and 2021, the earthquake in Haiti – just to name a few).

[2] Union Civil Protection Mechanism

Schultz, R. Frederking, et al. 2008) translate emergency calls from Spanish speaking users in the US, and *BabelDR* (Spechbach et al. 2019) is fixed-phrase translator, which has been recently proposed to facilitate communication between medical personnel and refugees. However, the problem of automatic speech translation between members of European cross-border FR teams implies several additional critical challenges: (i) it cannot afford erroneous translation even in highly noisy environments (Hunt et al. 2019); (ii) it must be multilingual (O'Brien, F. Federici, et al. 2018; O'Brien and F. M. Federici 2019) , and (iii) it must support both open domain speech and very specific jargon. A translation service of this kind can be projected as end-to-end speech-to-speech translation or as a pipeline configuration that consists of the automatic speech recognition–machine translation–text-to-speech modules. Since end-to-end translation does not offer transcriptions of the original speech or the translation protocols (both of which are highly valuable for verification of the accuracy of the translation and for mission debriefing sessions), we opt for a neural pipeline configuration. To facilitate the easy use of the application by FRs, it is embedded into the Telegram Messenger. Currently, the application covers speech translation between English, French, German, Spanish and Swedish and is embedded into the larger technology setup designed to support FRs, realized in the context of the European INGENIOUS project (`https://ingenious-first-responders.eu/`). The current version serves us as a baseline for a more advanced realization, which is under research.

OVERVIEW OF THE SPEECH TRANSLATION MODULE

Figure 1 shows the architecture of the proposed automatic speech translation application. As already mentioned above, due to the need for the transcriptions of the original audio in the source language $\mathcal{L}_1$ as well as the translations into the target language $\mathcal{L}_2$ in text format, the processing is separated into distinct steps, which are executed in sequence. The outputs of each step are made available to the next step and to the communication platform for logging. The first model in the pipeline is an Automatic Speech Recognition (ASR) module, which takes as input the $\mathcal{L}_1$ speech and transcribes it to text. The outcome is sent to the Machine Translation (MT) module, which translates it to $\mathcal{L}_2$; the translation is sent to the Text-to-Speech (TTS) model, which generates speech in $\mathcal{L}_2$.

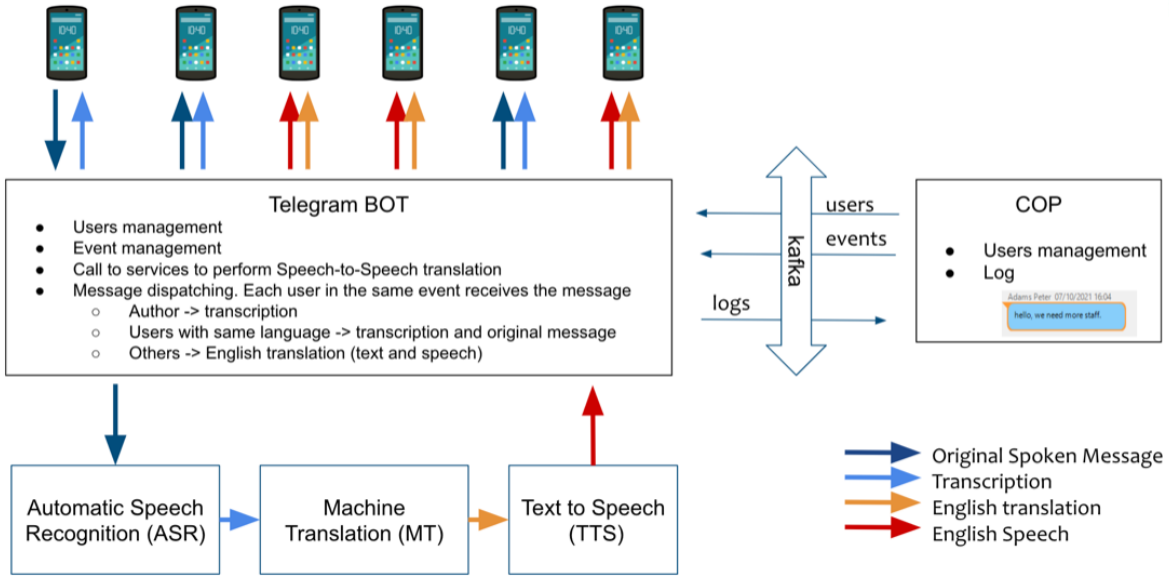

Figure 1. Architecture of the speech translation application

While the ideal setup of a speech-to-speech translation application might be envisaged as hands-free radio communication, our prototype implementation realizes smartphone-based communication through the Telegram Messenger.[3] The use of the Telegram application has several advantages. Firstly, it does not require additional hardware as part of the equipment of FRs. Secondly, it facilitates a straight-forward logging of transcripts and the translations. The management of the FR teams is performed centrally by the *Central Operations Platform* (COP), which also receives and logs transcripts of the communication. In addition, FRs receive a transcript of their own messages (e.g., for verification) as well as a transcript of the messages of the other communication parties (translated as needed), in addition to the spoken translated message.

In what follows, we briefly introduce each of the modules in our pipeline setup. For theoretical background on ASR, MT, and TTS the reader is asked to consult, e.g., (Jurafsky and Martin 2016).

[3] https://telegram.org/

Automatic Speech Recognition

The availability of data and schemes for training ASR models varies from language to language. In what follows, we describe the design and training schemes for the ASR models for the four languages covered in this work.

For **Spanish**, we benchmark two different ASR models: ConvGLU ASR (Zeghidour et al. 2018), which is entirely based on Convolutional Neural Networks (CNNs) (LeCun, Bengio, et al. 1995), and wav2vec2.0 (Baevski et al. 2020), a stack of CNN and Transformer (Vaswani et al. 2017) layers that is first trained on unlabeled speech and then fine-tuned to transcribed speech. ConvGLU ASR is trained with the wav2letter++/Flashlight toolkit (Pratap and Hannun 2018). It consists of a stack of CNN layers with gated linear units (GLU) (Dauphin et al. 2017). We use 191.42 hours of Spanish speech from the Common Voice dataset (Ardila et al. 2020) to train the model. This is a multilingual dataset, where users from around the world record their voices by pronouncing text prompted to them.

To train the model, we follow Flashlight's WSJ recipe (Collobert et al. 2016) using 80-dimensional log-mel spectrograms as input. The Auto Segmentation (ASG) criterion (Collobert et al. 2016) is used for optimization, with a batch size of 4 utterances. We decrease the learning rate in steps, setting it to 7.3 from epochs 0 to 34, then to 4.0 from epochs 34 to 85, and finally to 1.0 from epochs 85 to 129. Decoding is done via beam search with the output from the ConvGLU AM, altogether with a lexicon extracted from all the words in training and validation sets, plus a 4-gram LM trained on the training set with the KenLM framework (Heafield 2011).

Alternatively to the ConvGLU ASR, we train another model based on wav2vec2.0 LARGE, fine-tuning a model checkpoint that is open-sourced at HuggingFace (Wolf et al. 2020), which was pretrained with 21.4k hours of Spanish speech in the European Parliament, taken from the Voxpopuli dataset (C. Wang et al. 2021). The aim of this model is to explore whether a bigger scale in terms of model size and training data yields fewer transcription errors or not. This time, the input feature is the raw waveform, which is featurized by the wav2vec2.0 model. To compress this feature into text, we use three linear layers, fine-tuning them on the Common Voice training set with the Connectionist Temporal Classification (CTC) criterion (Graves et al. 2006) for optimization. Text is generated following a Byte Pair Encoding (BPE) of a 1000 unigrams, extracted from the text in the Common Voice training set. We use the SpeechBrain toolkit (Ravanelli et al. 2021) to perform the training, with a batch size of 4, a learning rate of 1.0 for the linear layers and a smaller learning rate of 0.0001 for the wav2vec2.0 encoder, as the latter is already pretrained and we do not encourage big modifications in its weights. Once we have fine-tuned the wav2vec2.0 model, we fine-tune it further, this time using background noises as data augmentation. We find it convenient to simulate the noisy environments where FR teams operate, so we use a varied set of sounds (fire, explosions, car engines, sirens, crowds, etc.) from the FSDnoisy18k (Fonseca et al. 2019) and UrbanSound8k (Salamon et al. 2014) datasets.

For **French** ASR, we train a large neural model that consists of 36 Transformer layers from scratch. For this purpose, we use 1,077 hours of French speech from the Multilingual LibriSpeech (MLS) dataset (Pratap, Xu, et al. 2020), which contains audiobook recordings. We follow the MLS recipe in Flashlight, with a batch size of 8 utterances, an input of 80-dimensional log-mel spectrograms and a learning rate of 0.02. SpecAugment (Park et al. 2020) is used for data augmentation, and AdaGrad (Lydia and Francis 2019) for optimization. The recipe provides a lexicon and a 3-gram Language Model (LM) that are pretrained with the training set and data from the Gutenberg Project (`https://www.gutenberg.org/`), which we use for beam search decoding of the output text.

Since the **German** ASR model is also Transformer-based, it is trained following the same Flashlight's MLS training recipe and design as the French model, in this case on 1,967 hours of speech. The training hyperparameters are identical to those of the French model; the LM is based on 5-grams, trained on the MLS training set plus books from the Gutenberg Project.

Lastly, the **Swedish** ASR is trained with the Flashlight toolkit, using another fast neural network design, based on time-depth separable (TDS) convolutions (Hannun et al. 2019). As we train with only 368 hours of Swedish speech from the NST dataset (National Library of Norway 2023), we find it convenient to avoid Transformer blocks here, as this type of architecture is more likely to excel with bigger amounts of data. The model is trained with a batch size of 8 utterances, a learning rate of 0.4, SGD optimizer and the CTC criterion. The lexicon is formed by a 10k unigram BPE, and we use a 4-gram LM trained with KenLM, both trained on the NST training set.

Machine Translation

For MT, we use the state-of-the-art open source ModernMT,[4] which, in addition to providing state-of-the art performance, is particularly well-suited for adaptation to specific application domains. This is of high relevance in

[4] https://github.com/modernmt/modernmt

our application due to the specific vocabulary and expressions used by FRs, which may not be well represented in general training corpora.

ModernMT uses neural machine translation (NMT) technology. The platform is designed to handle large volumes of text quickly and efficiently, with a focus on achieving high-quality translations that are tailored to specific domains. One of the key features of ModernMT is its ability to use domain-specific *Translation Memories* (TM)s to improve translation quality. A TM is a database that stores previously translated content, such as documents, sentences, or phrases, and their corresponding translations. ModernMT can use these TMs to help improve the accuracy and consistency of its translations.

When translating text, ModernMT searches the TM for matches to the text being translated. If a match is found, the corresponding translation is used to inform and improve the translation of the new text. This process is particularly effective in domain-specific translations, where the used language may be highly specialized or technical. By leveraging domain-specific TMs, ModernMT can produce more accurate and consistent translations that are better suited to the specific needs of a particular industry or field.

We trained models for the language pairs we worked on using freely available parallel corpora from Opus[5], which is a large unified resource that helps access a large number of corpora for a wide variety of languages. We then enhanced the (generic) translation models with translation memories to add domain-specific vocabulary and avoid mistranslations that appeared when relying on the generic model only.

Text-to-Speech

We use a neural TTS model to speak the output in $\mathcal{L}_2$ provided by the MT model. Specifically, we use the Tacotron2 model (Shen et al. 2018), whose design consists mainly of a linguistic features encoding part, and a decoding side based on attention and recurrent neural networks. Firstly, the input text is encoded by a stack of convolutional and long-short term memory (LSTM) layers (Hochreiter and Schmidhuber 1997). Then, the output mel-spectrogram signal is decoded autoregressively through an LSTM stack by applying the attention mechanism to the encoded text at every step. The autoregressive generation is stopped when a *STOP* token is predicted.

Originally, Tacotron2 used a WaveNet module (Oord et al. 2016) to vocode the output mel-spectrogram into a waveform that can be played through speakers; some newer open source versions use MelGAN (Kumar et al. 2019) as a vocoder to reduce synthesis latency. We have replaced MelGAN by a pretrained Multi-Band MelGAN (MB-MG) model (G. Yang et al. 2021), which according to our benchmarking, performs 55% faster than its predecessor. This allows for faster response on the field, which is very important for emergency teams. Furthermore, we have introduced two more modifications to the Tacotron2 model. Firstly, we have introduced the Double Decoder Consistency technique (Gölge E. 2020), which helps the attention mechanism in the decoder to achieve a more accurate alignment between the input text and the outgoing speech. This is especially important when users want to synthesize longer sentences, which in our use case means that the FR teams are able to speak for a longer time without major impacts in the speech synthesis. Secondly, we found that Tacotron2 introduces some uncontrollable random variations on speech, even for the same input. To solve this problem, we changed the dropout in the PreNet module of Tacotron2, which was causing this behavior, replacing it by batch normalization.

The entire TTS model was trained on the open-access LJSpeech dataset, which consists of 24 hours of audiobook speech from a female speaker.

ILLUSTRATION OF THE APPLICATION OF THE SPEECH TRANSLATOR

The prototype version of our speech translator can be easily used on any cell phone (Android or iOS) using the Telegram Messenger, as well as on tablets or computers (see Figure 2). Once the user connects to the translation bot and is registered in the COP, they can send text and voice messages, which are received by the listeners (determined via the configuration in the central platform). Users receive messages sent by other participants according to their language setting, with written and spoken translations.

Additional functionality of the app includes the configuration of the user's language, the list of users assigned to the same event, a history of messages, etc.

A transcript of the communications is additionally available in the COP platform (as seen in Figure 3), translated into the centrally configured language (in this case, English).

[5]https://opus.nlpl.eu/

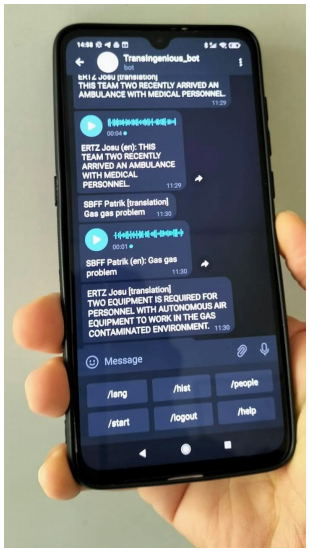

Figure 2. Use of the translation application on a mobile phone

Table 1. WER for the proposed Spanish ASR systems, evaluated with the Spanish Common Voice dataset.

	Valid WER (%)	*Test WER (%)*	*Noisy Valid WER (%)*	*Noisy Test WER (%)*
ConvGLU	33.32	36.69	-	-
wav2vec2.0	11.67	12.85	18.45	19.37
+ *noise augmentation*	**10.27**	**11.29**	**13.97**	**15.09**

PRELIMINARY PARTIAL EVALUATION

In addition to the usual task-specific quantitative evaluations detailed below, the presented speech translation application has been assessed in a series of practice-oriented trials carried out in the context of the global INGENIOUS technology setup for FRs, of which this application forms part.

Our quantitative evaluation focused so far on ASR and on MT for ES-EN, FR-EN, and DE-EN. For the pending evaluation of TTS, qualitative questionnaires will be used.

ASR Evaluation

We use Word Error Rate (WER) to measure the quality of ASR transcriptions. WER is defined as the sum of word substitutions, deletions and insertions done by the ASR, with respect to the ground truth transcription, divided by the number of words in the latter.

Table 1 shows the quality of the two Spanish ASR models, ConvGLU and wav2vec2.0 (W2V2), evaluated on the Common Voice validation (*Valid*) and test (*Test*) sets. The higher amount of data used for training wav2vec2.0, plus its Transformer-based design with more parameters (317M versus 40M in ConvGLU), makes this model score more than 22% higher in terms of WER. This significant impact on the performance encouraged us to develop French and German Transformer-based models, instead of convolutional ones. French ASR achieves 24.52% WER on the MLS validation set and a 25.79% WER on the test set. For German ASR, the MLS validation set has been transcribed with a 19.82% WER and the test set with a 21.14% WER. As previously mentioned, we still use a convolutional architecture for Swedish due to the lower amount of data available for it, as Transformer models are known to be more reliant on big amounts of data. Still, we achieve a 19.95% WER and a 23.99% WER on NST validation and test sets, respectively.

Furthermore, we also compare our Spanish wav2vec2.0 model with the model that was fine-tuned with noise augmentations, to check whether the latter has any advantages in highly noisy scenarios as in our use case. For such a task, we do not only check WER scores with *Valid* and *Test* sets, but we also corrupt these datasets with background noises, creating noisy versions of them (*Noisy Valid* and *Noisy Test*). As we expected, the model that was fine-tuned with noise augmentations is more robust towards noise, reducing the WER by more than 4% for the noisy sets. Furthermore, even for clean audio, the noise augmentation version improved the WER by > 1%. This

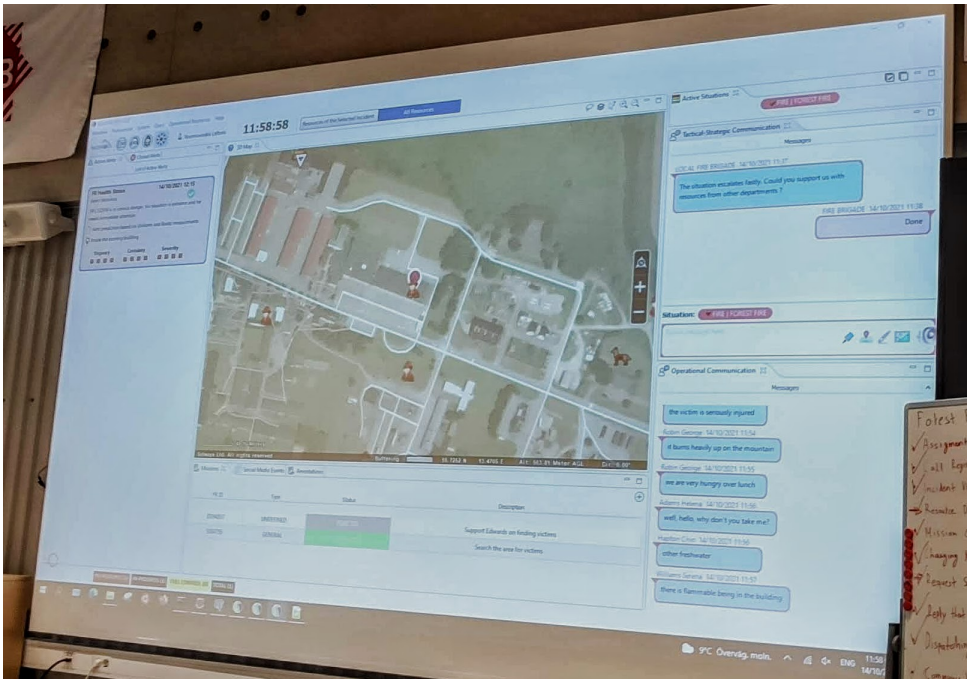

Figure 3. Transcript of communications in the Central Operations Platform

Table 2. Evaluation of the MT component for some language pairs

		BLEU (%)	*Post-Edit (%)*
ES–EN	*ModernMT (ours)*	**56.19**	**72.35**
	Google	44.80	65.03
FR–EN	*ModernMT (ours)*	**42.39**	**59.34**
	Google	36.72	55.26
DE–EN	*ModernMT (ours)*	**35.09**	**58.44**
	Google	31.52	55.09

shows that noise augmentation is a technique with a significant impact on the performance of ASR operating in noisy scenarios, motivating its usage for the other languages in the next iterations as well.

MT Evaluation

As mentioned above, the applied MT module uses an Open Source framework, with models trained on freely available data. As such, the evaluation of its performance is not the main focus of this work. However, to ensure that we obtain results that are close to the state-of-the-art, Table 2 presents the figures for some of the language pairs we worked with.

The presented numbers are the widely used BLEU score values[6], and the Matecat Post-Edit score values[7], calculated over a set of 2,000 sentences. Both BLEU and the Post-Edit compare each translated sentence to the corresponding correct reference sentence, with a higher score meaning that the automatic translation is closer to the reference and thus assumed to be a better translation.

For evaluation, a few thousand sentences were initially removed from the training corpus (and thus not used for training). Each sentence of the held-out set was manually translated into the target language; the translations were used as the ground truth test set.

In all cases, our ModernMT models are the clear winners, even with better results than Google Translate. However, it is to be noted that the evaluation is done on sentences that are similar in style and domain to the training sentences

[6]https://en.wikipedia.org/wiki/BLEU

[7]https://github.com/modernmt/modernmt/wiki/FAQs#q-how-does-the-matecat-post-editing-score-that-is-output-from-the-mmt-evaluate-command-work

(both are from the MultiUN and OpenSubtitles corpora), giving our system an advantage over Google Translate, which is trained on other (and much larger) resources. Still, it is a strong indication of the competitiveness of our models.

As also mentioned above, one of the positive aspects of the MT module is that it facilitates the addition of user-provided translations. This allows the inclusion of domain-specific vocabulary that is not represented in the training data used to produce the translation models, as well as to improve translations for the specific application context and thus fix "mistranslations" produced by the underlying "generic" model. One such mistranslation seen by users during a trial was, e.g., the translation of the Spanish *equipo de extracción* 'extraction team' as 'mining team' (where the non-adapted model misinterpreted *extracción* as mining of minerals,[8] instead of the extraction of victims), which is easily fixed by adding the correct translation. Such domain adaptation proved to be very valuable, in particular with the possibility for end users to add their own vocabulary and translations at runtime without requiring the knowledge and technical skills to train and deploy a translation model from scratch. Due to the limited amount of data available for these cases it was not possible, however, to statistically evaluate the impact of this adaptation.

THE STATE OF THE WORK

Processing speed and quality of the results of the presented speech translation prototype make it a valuable means for bridging the communication gap between teams speaking different languages already in its current state. As a rule, translation is reliable and can be tuned by adding specific vocabulary and translations to avoid mistranslations. The TTS output is intelligible and of sufficient quality to be well understood in the field. The most critical component is the automatic speech recognition, as any error in transcribing the input speech inevitably leads to errors in the subsequent steps. While this component has improved significantly in its latest iteration, errors remain unavoidable (even a human listener will not be able to correctly transcribe all utterances in all situations). It can therefore be important for the sender of the message to review the correctness of the transcription (a transcript of their own message is provided to them).

Language coverage is one of the limitations of the system, as individual models need to be trained for transcribing each input language, synthesizing each output language, as well as to be translated from the input language into all output languages. New developments with multilingual models that cover a large variety of languages within a single model instead of requiring a separate model for each language are promising for breaking this barrier in the future. However the computational needs to create such a model remain huge.

For ASR, multilingual models such as the recently released Whisper model (https://openai.com/blog/whisper/) could be a significant step in that direction, since they are trained on a significant amount of multilingual speech data (Whisper, e.g., has been trained on 680 hours). It is part of our future work to research how multilingual models respond to our use case, to check if it would still be better to fine-tune them separately for each language and, especially, to what extent multilingual models can handle the noise of our use case. Furthermore, we need to assess how such models handle audio distortions introduced by compression codes in the communication channels.

For machine translation, while resources for training many language pairs are available, and pretrained models for many languages have also become more widely available, there is no publicly accessible multilingual (locally deployable) system that would provide the same flexibility for domain-adaptation as our current solution. More basic research is needed on this topic.

Speech synthesis also lacks good models (or any models) for less widely used (especially non-European) languages, and in many cases, there is not sufficient training data available even to train new specific models for these languages. New approaches need to be researched that do not require large volumes of training data.

To summarize, in its current version, the presented speech translator is a prototype implementation, which, while operational and usable, is not yet a mature product. In our continuing research, we aim to further improve it. Apart from the above limitations concerning, in particular, ASR and MT, from the practical side, it would be necessary to integrate the functionality with the communication systems already in use instead of relying on a separate mobile application. The design of the speech translator makes the integration very straight-forward. Furthermore, while the prototype implementation relies on the internet for communication between users as well as with the underlying translation service, it would be perfectly feasible to have a mobile deployment in the field, along with the other communication infrastructure used in such situations and not depend on remote access to the translation service.

[8] https://dictionary.cambridge.org/dictionary/spanish-english/extraccion

ACKNOWLEDGEMENTS

The work described in this paper has been funded by the European Commission in the framework of its Horizon 2020 R&D Program under the contract number 833435.

REFERENCES

Ardila, R., Branson, M., Davis, K., Kohler, M., Meyer, J., Henretty, M., Morais, R., Saunders, L., Tyers, F., and Weber, G. (2020). "Common Voice: A Massively-Multilingual Speech Corpus". In: *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 4218–4222.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. (2020). "wav2vec 2.0: A framework for self-supervised learning of speech representations". In: *Advances in Neural Information Processing Systems* 33, pp. 12449–12460.

Collobert, R., Puhrsch, C., and Synnaeve, G. (2016). "Wav2Letter: an End-to-End ConvNet-based Speech Recognition System". In: *CoRR* abs/1609.03193. arXiv: 1609.03193.

Dauphin, Y. N., Fan, A., Auli, M., and Grangier, D. (2017). "Language modeling with gated convolutional networks". In: *International conference on Machine Learning*. PMLR, pp. 933–941.

Fonseca, E., Plakal, M., Ellis, D. P., Font, F., Favory, X., and Serra, X. (2019). "Learning sound event classifiers from web audio with noisy labels". In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 21–25.

Gölge E. (2020). *Solving Attention Problems of TTS models with Double Decoder Consistency*.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks". In: *Proceedings of the 23rd International Conference on Machine Learning*, pp. 369–376.

Hannun, A., Lee, A., Xu, Q., and Collobert, R. (2019). "Sequence-to-Sequence Speech Recognition with Time-Depth Separable Convolutions". In: *Proc. Interspeech 2019*, pp. 3785–3789.

Heafield, K. (2011). "KenLM: Faster and Smaller Language Model Queries". In: *Proceedings of the Sixth Workshop on Statistical Machine Translation*. WMT '11. Edinburgh, Scotland: Association for Computational Linguistics, pp. 187–197.

Hochreiter, S. and Schmidhuber, J. (1997). "Long short-term memory". In: *Neural Computation* 9.8, pp. 1735–1780.

Hunt, M., O'Brien, S., Cadwell, P., and O'Mathúna, D. P. (2019). "Ethics at the intersection of crisis translation and humanitarian innovation". In: *Journal of Humanitarian Affairs* 1.3, pp. 23–32.

Jurafsky, D. and Martin, J. (2016). *Speech and Language Processing, 2nd Edition*. Upper Saddle River, NJ: Prentice Hall.

Kumar, K., Kumar, R., De Boissiere, T., Gestin, L., Teoh, W. Z., Sotelo, J., Brébisson, A. de, Bengio, Y., and Courville, A. C. (2019). "Melgan: Generative adversarial networks for conditional waveform synthesis". In: *Advances in Neural Information Processing Systems* 32.

LeCun, Y., Bengio, Y., et al. (1995). "Convolutional networks for images, speech, and time series". In: *The Handbook of Brain Theory and Neural Networks* 3361.10, p. 1995.

Lydia, A. and Francis, S. (2019). "AdaGrad—an optimizer for stochastic gradient descent". In: *Int. J. Inf. Comput. Sci* 6.5, pp. 566–568.

Nallasamy, U., Black, A. W., Schultz, T., Frederking, R., and Weltman, J. (2008). "Speech Translation for Triage of Emergency Phonecalls in Minority Languages". In: *Coling 2008: Proceedings of the workshop on Speech Processing for Safety Critical Translation and Pervasive Applications*, pp. 48–53.

Nallasamy, U., Black, A. W., Schultz, T., and Frederking, R. E. (2008). "NineOneOne: Recognizing and Classifying Speech for Handling Minority Language Emergency Calls." In: *LREC*.

National Library of Norway (2023). *The Norwegian Language Bank*.

O'Brien, S., Federici, F., Cadwell, P., Marlowe, J., and Gerber, B. (2018). "Language translation during disaster: A comparative analysis of five national approaches". In: *International journal of disaster risk reduction* 31, pp. 627–636.

O'Brien, S. and Federici, F. M. (2019). "Crisis translation: Considering language needs in multilingual disaster settings". In: *Disaster Prevention and Management: An International Journal* 29.2, pp. 129–143.

Oord, A. van den, Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. (2016). "WaveNet: A Generative Model for Raw Audio". In: *9th ISCA Speech Synthesis Workshop*, pp. 125–125.

Park, D. S., Zhang, Y., Chiu, C.-C., Chen, Y., Li, B., Chan, W., Le, Q. V., and Wu, Y. (2020). "SpecAugment on large scale datasets". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 6879–6883.

Pratap, V. and Hannun, A. (2018). "wav2letter++: The fastest open-source speech recognition system". In: *CoRR, vol. abs/1812.07625*.

Pratap, V., Xu, Q., Sriram, A., Synnaeve, G., and Collobert, R. (2020). "MLS: A Large-Scale Multilingual Dataset for Speech Research". In: *Proc. Interspeech 2020*, pp. 2757–2761.

Ravanelli, M., Parcollet, T., Plantinga, P., Rouhe, A., Cornell, S., Lugosch, L., Subakan, C., Dawalatabad, N., Heba, A., Zhong, J., et al. (2021). "SpeechBrain: A general-purpose speech toolkit". In: *arXiv preprint arXiv:2106.04624*.

Salamon, J., Jacoby, C., and Bello, J. P. (2014). "A dataset and taxonomy for urban sound research". In: *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 1041–1044.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., et al. (2018). "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 4779–4783.

Spechbach, H., Gerlach, J., Karker, S. M., Tsourakis, N., Combescure, C., Bouillon, P., et al. (2019). "A speech-enabled fixed-phrase translator for emergency settings: Crossover study". In: *JMIR medical informatics* 7.2, e13167.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). "Attention is all you need". In: *Advances in Neural Information Processing Systems* 30.

Wang, C., Riviere, M., Lee, A., Wu, A., Talnikar, C., Haziza, D., Williamson, M., Pino, J., and Dupoux, E. (2021). "VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 993–1003.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2020). "Transformers: State-of-the-art natural language processing". In: *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pp. 38–45.

Yang, G., Yang, S., Liu, K., Fang, P., Chen, W., and Xie, L. (2021). "Multi-band MelGAN: Faster waveform generation for high-quality text-to-speech". In: *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, pp. 492–498.

Zeghidour, N., Xu, Q., Liptchinsky, V., Usunier, N., Synnaeve, G., and Collobert, R. (2018). "Fully convolutional speech recognition". In: *arXiv preprint arXiv:1812.06864*.