Removing European Language Barriers with Innovative Machine Translation Technology

Dario Franceschini, Chiara Canton, Ivan Simonini (PerVoice),
Armin Schweinfurth, Adelheid Glott (alfatraining),
Sebastian Stüker, Thai-Son Nguyen, Felix Schneider, Thanh-Le Ha, Alex Waibel (KIT),
Barry Haddow, Philip Williams, Rico Sennrich (UEDIN)
Ondřej Bojar, Sangeet Sagar, Dominik Macháček, Otakar Smrž (CUNI),

PerVoice, Italy; name.surname@pervoice.it
alfatraining; name.surname@alfatraining.de
Karlsruhe Institute of Technology, Germany; name.surname@kit.edu
University of Edinburgh (UEDIN)
Charles University, MFF ÚFAL (CUNI); surname@ufal.mff.cuni.cz

Abstract

This paper presents our progress towards deploying a versatile communication platform in the task of highly multilingual live speech translation for conferences and remote meetings live subtitling. The platform has been designed with a focus on very low latency and high flexibility while allowing research prototypes of speech and text processing tools to be easily connected, regardless of where they physically run. We outline our architecture solution and also briefly compare it with the ELG platform. Technical details are provided on the most important components and we summarize the test deployment events we ran so far.

Keywords: automatic speech recognition, spoken language translation, machine translation, automatic minuting, live transcription, live translation

1. Introduction

While natural language processing (NLP) technologies like automatic speech recognition (ASR), machine translation (MT), spoken language translation[1] (SLT), natural language understanding (NLU), or automatic text summarization have recently seen tremendous improvements, and are provided to end users as services by large companies like Google, Microsoft or Facebook,[2] the output quality of applications is still insufficient for practical use in daily communication. The goal of the ELITR (European Live Translator) project[3] is to advance and combine different types of NLP technologies to create end-to-end systems that are usable in serious business communication. Specifically, the ELITR project targets the advancement and application of ASR and SLT in two challenging settings:

- Face-to-face conferences (interpreting official speeches and workshop-style discussions)
- Remote conferences (interpreting discussions held over a on-line platform)

In addition to addressing technological challenges in ASR, SLT, and MT, the project covers a large number of languages: ELITR tests its ASR technology in 6 EU languages. The subsequent MT technology is currently able to translate among all 24 official EU languages but aims at supporting a larger set of language relevant for our user partner, the languages of members of European Organisation of Supreme Audit Institutions, EUROSAI.[4]

The paper is structured as follows: In Section 2., we describe the core of our systems, the processing platform, which is used in both face-to-face and remote meetings settings. In Section 3. we go through some of the differences between ELITR platform and the ELG Grid. In Section 4., we summarize the design decisions and status of the technologies connected to the platform. Section 5. describes our field tests and our first experience.

2. Processing Platform

The architecture of ELITR SLT systems builds upon the PerVoice Service Architecture, a proprietary software solution with roots supported also by several previous EU projects.
This architecture is composed of a central unit called the Mediator, and several modules for processing pipelines, called Workers, which can be easily provided by universities or research labs. These Workers are implemented as standalone programs that connect to the Mediator via TCP/IP. The communication protocol (or API) is prescribed and among other things requires each Worker to indicate the service it provides, for instance translation from a given source to a given target language. Typically, Workers are

[1] We interpret this term in the narrow sense: speech in one language to text in another language

[2] Microsoft Translator translates between 62 languages, with 22 handled by the novel neural approach, and recognizes speech in 11 languages. Two variants of Chinese and English can be included in a customized component.

[3] http://elitr.eu/

[4] EUROSAI languages are all EU languages and Albanian, Arabic, Armenian, Azerbaijani, Belorussian, Bosnian, Georgian, Hebrew, Icelandic, Kazakh, Luxembourgish, Macedonian, Moldovan, Montenegrin, Norwegian, Russian, Serbian, Turkish, and Ukrainian, over 40 languages in total.

simple wrappers of the partners' respective tools or research prototypes.
Clients connect to the Mediator, requesting a particular type of output and providing a source data stream, i.e. audio or text based on their use cases. The Mediator orchestrates the service provision by contacting the required Workers. PerVoice Service Architecture supports both batch and real-time processing.

2.1. Metadata: Fingerprint and Types

The first problem addressed by the PerVoice Service Architecture is the declaration of Services and service requests descriptions. For this purpose, so called *fingerprints* and *types* are used to specify the exact language and genre of a data stream. Fingerprints consist of a two-letter language code (ISO369-1) followed by an optional two-letter country code (ISO3166) and an optional additional string specifying other properties such as domain, type, version, or dialect (`ll[-LL[-dddd]]`). Types are: audio (audio containing speech), text (properly formatted textual data), unseg-text (unsegmented textual data such as ASR hypotheses).
Service descriptions and service requests are fully specified by their input and output fingerprints and types. For example, the ASR service which takes English audio as input and provides English unsegmented text adapted on news domain will be defined by "`en-GB-news:audio`" input fingerprint and "`en-GB-news:unseg-text`" output fingerprint. The service request of German translation of English audio will be defined by "`de-DE-news:audio`" input fingerprint and "`en-GB-news:text`" output fingerprint.

2.2. Workflow

When a Worker (the encapsulation of a service) connects to the Mediator (orchestration service) on a pre-shared IP address and port, it declares its list of service descriptions, i.e. the list of services it offers. As soon as the connection is established, the Worker waits until a new service request is received.
Several Workers can connect to the Mediator and offer the same service, which allows for a simple scaling of the system. As soon as the new service request has been accepted, the Worker waits for incoming packets from the Client's data stream to process, and performs specific actions depending on the message types (data to be processed, errors, reset of the connection). When the Client has sent all the data, the worker waits until all pending packets have been processed, terminates the connection with the Client and waits for a new Client to connect.
From the Client perspective, when a Client connects to the Mediator, it declares its service request by specifying which kind of data it will provide (output fingerprint and type) and which kind of data it would like to receive (input fingerprint and type). If the Mediator confirms that the mediation between output type and input request is possible, the Client starts sending and receiving data. When all data has been sent, the Client notifies it to the Mediator and waits until all the data has been processed by the Workers involved in its request. The Client can then disconnect from the Mediator.

2.3. Mediation

In order to accomplish a Client's request, a collection of Workers able to convert from the Client's output fingerprint and type to the requested input fingerprint and type must be present. For example, if a Client is sending an audio stream with the fingerprint `en-GB-news:audio` and requests `en-GB-news:unseg-text`, the Mediator must find one Worker or a concatenation of multiple Workers that are able to convert audio containing English into unsegmented English text, i.e. a speech recognition Worker in the example. The Mediator searches for the optimal path to provide a service using a best path algorithm that works on fingerprint names and types match.
In order to make sure that a mediation is still possible even if there are no workers available matching the requested stream types and fingerprints, back-up strategies have been implemented, which relax the perfect match on country and domain fingerprint's section.

2.4. MCloud Library

Through its light-weight API MCloud, the PerVoice Service Architecture defines a standard for services integration, allowing different partners integration and a flexible usage for different use cases. The Mediator supports parallel processing of service requests in a distributed architecture.
MCloud is a C library which implements the raw XML protocol used by the PerVoice Service Architecture and exposes a simplified API for the development of Clients and Workers. For convenience, the library integrates some high-level features like audio-encoding support and data package management. A .NET and a Java wrapper of the MCloud API are available in order to support the development of client desktop applications for the PerVoice Service Architecture.

3. Comparison of ELITR and ELG Platforms

Another EU project, European Language Grid (ELG)[5] also develops a common platform for natural language processing.
While starting from similar intentions, ELITR and ELG focus on different use cases. ELITR targets real-time business use cases—like face-to-face and remote video conferencing for selected events—ELG focuses on the creation of a shared European Language Technologies catalogue and marketplace for self-service usage of provided technologies. Both purposes and intentions are valuable but result in different technological approaches.
ELITR use cases include live video streaming and automatically transcribed and translated subtitles. For this reason the project preferred the low-latency solution provided by the PerVoice Service Architecture, which works in real-time and also enables the transparent concatenation of services (e.g., ASR output passed as input to translation Worker) based on "on-air" services. Real-time communication is provided by a fast protocol working over TCP/IP

[5] `https://www.european-language-grid.eu/`

sockets which ensures smaller latencies in contrast to approaches relying on external message brokers that introduce asynchronous interaction and delays.
The decentralized approach of the PerVoice Service Architecture allows companies to avoid sharing proprietary technologies. Furthermore, the actual service provider of a Worker component is secondary to the actual functionality being provided. ELG instead prefers the service categorization approach, creating a catalogue of services deployed in its infrastructure.
The ELITR solution could be deployed offline, should the use case require special security and data privacy measures—assuming that there are sufficient hardware resources and a partner agreement. The ELG grid instead is deployed only in cloud.
In general, we highlight the fact that language technologies can rely on different software architectures, and not all of them are suitable to be containerized. For example, a complex language processing solutions could run more than one process, making it harder to manage the container and debug problems, or they could have high resource requirements. Large virtual machine images become an issue when thousands of containers need to be deployed across a cluster. The PerVoice Service Architecture instead delegates service management to individual parties contributing services to the infrastructure, in order to exploit their specific training and knowledge of the technologies and systems for a better resource allocation and usage.

4. ELITR Technologies

With respect to the core language processing technology needed to realize the simultaneous translation service presented here we face several research questions that need to be addressed. Besides the obvious challenge of providing speech translation with sufficient performance, the special case of simultaneous speech translation for conferences, talks and lectures brings specific challenges with it. Two foremost challenges are a) that speech translation has to happen in real-time and with low latency in order to be simultaneous, and b) to cover and adapt to a large variety of domains as the topics of talks and conferences can be virtually arbitrary; therefore systems need to be either domain-independent (a still unsolved research question) or need to be able to adapt to the current domain, autonomously or with as little human supervision as possible.
Currently the systems for speech translation also undergo an architecture transformation from statistical models based on Bayes' rule towards all neural models that give better performance. In our scenario this transformation has to be done under the aspects of the need for low latency translation which leads to task specific considerations.

4.1. Architecture Consideration

Over the last years the basic technology of the components for speech translation has undergone radical transformations. While for decades systems for speech recognition and machine translation where based on Bayes' rule and made use of statistical methods such as Hidden Markov Models, Gaussian Mixture Models, N-Gram Models, and Phrase Based Translation Models, lately the use of neural networks has led to significantly improved performance. While first individual components, such as the acoustic model or the language models, of the systems were replaced, the latest improvements were gained from end-to-end systems that solve the problem of automatic speech recognition, machine translation etc. with a single neural network architecture, instead of solving the problem with several models given by Bayes' rule.
This single network architecture can go to the extreme of solving the whole problem of speech translation with one single neural network architecture.

4.1.1. Current SLT Architecture in ELITR

At this time, end-to-end speech translation systems do not yet outperform cascaded systems consisting of several components (Niehues et al., 2019). End-to-end speech recognition models (Nguyen et al., 2019; Pham et al., 2019) have been showing promising performance but have limit when being used in online conditions. Therefore, in ELITR we use a cascaded speech translation system consisting of:

- Automatic Speech Recognition System (ASR)
- Punctuation System (PUNCT)
- Machine Translation System (MT)

Automatic Speech Recognition In our system, the ASR component is in charge of processing the audio stream sent from recording clients and output a stream of text transcript to the next component in the pipeline. We currently follow the HMM/ANN hybrid approach (Fügen et al., 2008; Niehues et al., 2018) to build up the ASR model. In this approach, ASR modeling is handled by two separate components: acoustic model (AM) and language model (LM). The task of AM is to model acoustic observations with regard to the labels of context dependent phonemes. As recent advances in the field of ASR, deep neural networks are used to leverage the modeling capacity of the AM on many hours of speech training data. Separately from AM, LM is trained solely on text data and it is used to provide the probabilities of word sequences. The AM and LM are then used in a dynamic decoding framework that is capable of online and low-latency inference. As one of the most important advantages of the hybrid approach, both AM and LM can be easily adapted for better performance if in-domain data is available for a particular application setup.

Punctuation System The hypotheses from speech recognition contain no punctuation. As our machine translation system is trained on well-structured, written sentence-level texts, we use a separate component to insert punctuation and sentence boundaries into the ASR output. This component also adds correct capitalization to the otherwise lowercased hypotheses.
Essentially, the punctuation system is a monolingual translation system, which translates the lower-cased, unsegmented outputs from the ASR components into well-formed texts prior to the translation system (Cho et al., 2015). We can employ any kind of translation approach and it is only required to train on a small amount of monolingual data. In our current punctuation system, for each language, we train a neural model on spoken texts, e.g the

transcripts of TED talks. Using our compact representation described by Cho et al. (2017), we are able to add punctuation and correct capitalization in one go. Furthermore, this compact representation helps to reduce the vocabulary size of our neural-based monolingual system, thus, reducing the model size and making the training of such system faster.

Machine Translation System With the ultimate goal of featuring a translation system for all EUROSAI languages, we opt for the multilingual approach (Ha et al., 2016; Ha et al., 2017; Johnson et al., 2017) where a single system is able to translate from and to multiple languages. This approach has many advantages:

- It leverages the large availability of multi-way, multilingual corpora in European languages such as the corpus of European Parliament documents and speeches' transcription (Europarl) (Koehn, 2005), the collection of legislative texts of the European Union (JRC-Acquis) (Steinberger et al., 2006) or the texts extracted from the document of European Constitution (EUconst) as well as the WIT[3] corpus extracted from TED talks (TED) (Cettolo et al., 2012).

- It uses the multilingual information to help improve the translation of the language pairs which are considered as low-resource languages in some domains. Our research has shown that our multilingual translation system maintains parity with the translation quality of systems trained on individual language pairs on the same small amount of data.

- In practice, having a small number of multilingual systems to cover all language pairs significantly reduces the development and deployment efforts compared with having one system for each pair.

Our multilingual systems are based on the neural sequence-to-sequence with attention framework (Bahdanau et al., 2014) and shares the internal representation across languages (Pham et al., 2017). At present, we have one many-to-many Transformer model (Vaswani et al., 2017) providing translation between all pairings of 36 languages, along with several specialized models focused on subsets of languages, in particular the project's primary languages of English, Czech, and German, see i.a. (Popel and Bojar, 2018; Popel et al., 2019).
The resulting multilingual models after training can be used immediately in deployment or can go through a language adaptation step. This language adaptation is simply continuing training the multilingual model on the data of a specific language pair for a few epochs in order to improve the individual translation performance. While we need to do this language adaptation for every single language pair in our system, it is a trivial job since we could automate the process with the same settings and it takes only a little of time and computing resources to reach decent performances.

4.2. Low-Latency Speech Translation

In order to realize low latency in automatic speech recognition we work with speculative output. The decoder in our speech recognition system realizes a Viterbi beam search. Due to the beam, partial hypotheses often have a stable part in which all alternative hypotheses have been pruned away by the beam further ahead in the search, and an unstable part that contains several competing hypotheses that fall within the beam.
Therefore it is possible to output the stable part, knowing that it will never change again as the search progresses. Previous experiments have shown that such a strategy would lead to a latency of about 6–8 seconds. A user study had shown that this considered too high a latency by the users. We therefore lowered the latency further by using speculative output, always putting out the current best hypothesis. Often this hypothesis will stay the most likely hypothesis, as the search progresses. In case it changes, we make use of an update mechanism that allows us to update the recent part of the hypothesis as necessary.
The punctuation component is set to generate the segmented, well-formed text whenever it receives any output, either unstable or stable, from the speech recognition system. And it passes its outputs along with the information of stability to the machine translation component.
Normally the machine translation component waits for the whole sentence before conducting the translation process. To reduce the latency, we force the component to directly and constantly produce outputs right after it receives outputs of the punctuation component. It might then fix the generated translation to be stable by its best hypothesis.
This brings down the average word-based latency, i.e. the time from which the last word of the sentence was spoken until the translation of that sentence is displayed and never changed again by the update mechanism, to under 5 seconds.

5. Practical Tests

While each of the components (ASR, punctuation, MT) are tested and evaluated on their own, on their respective test sets, the whole complex setup also has to be evaluated.
We are still working on a tool which would allow for a rigorous evaluation of the performance considering multiple aspects like translation quality, delay or text updates which may damage the end user experience.
For the time being, we focus on running many 'field tests', deploying the technology at various occasions. Our experience in the two intended settings (face-to-face multilingual conferences and remote conferencing) is described in the respective sections below.

5.1. Tests of Multi-Target Conference Speech Translation

Since the ELITR kick-off in January 2019, we carried out several tests and dry-runs to present our live-subtitling system. It first started with a Students Firms Fair in March 2019. During this event, we provided live subtitles on different languages that were spoken on the presentation stage, and we also collected a rather challenging speech test set (Macháček et al., 2019) which serves in the Non-native SLT task at IWSLT 2020.[6]

[6] http://workshop2020.iwslt.org/doku.php?id=non_native_speech_translation

Next, we had two officially planned events organized by the Supreme Audit Office of the Czech Republic (SAO) that were held in June 2019 and October 2019. In these events, the subtitles were delivered live to the participants through the presentation platform on their laptops. Apart from this, we also tested the input from interpreters into Czech and English respeakers. We also tried to show the live translation of the speaker in Czech, Hungarian, Spanish, German and Dutch from English. These translations were, however, unstable and inconvenient for users to interpret the context of the discussion. This event highlighted the required scope for improvement both in service functionalities and user experience. We made many critical observations from these two events and we gradually improved several aspects of the system for another dry-run in February 2020. Apart from the usual two-line subtitle view, we now present also a paragraph view of the transcript which contains more text in a history-style view. The subtitles were presented in English and translated into German, Czech, Russian, French, Hungarian, Polish, and Dutch.

5.2. Tests of Remote Conferencing

The functionality of live transcription has been succesfully tested in the field of labour market training by alfatraining, an educational provider using alfaview®.[7] A remote call participant with hearing impairment used the live transcript to follow the lessons and participate in discussions with a lecturer and other participants.

In another test, CUNI organized a call between two persons. One person followed only the transcript or translation, without listening. The second person was describing a word without saying it explicitly. We showed on multiple person pairs and languages that it is possible to guess the explained word both from transcripts and automatic translations of natural, spontaneous speech.

6. Conclusion

The PerVoice Service Architecture decouples clients and service providers by providing a simple protocol and an integration library, available for the major platforms, to connect both end-user application and service engines to it. It simplifies the creation of workflows among different service providers by providing automatic workflow creation solution.

Populated with state-of-the-art systems for automatic speech recognition and machine translation developed at KIT, UEDIN and CUNI, the architecture proves its applicability in challenging settings, as needed by the EU project ELITR.

Tests showed practical usability of our systems for face-to-face and remote conferences in real conditions. They also showed that the current and future main challenge is to improve speech recognition, especially for non-native dialects and out-of-vocabulary words.

Acknowledgement

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 825460 (ELITR).

[7] https://alfaview.com

7. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR*, abs/1409.0473.

Cettolo, M., Girardi, C., and Federico, M. (2012). Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy, May.

Cho, E., Niehues, J., Kilgour, K., and Waibel, A. (2015). Punctuation Insertion for Real-time Spoken Language Translation. In *Proceedings of the Eleventh International Workshop on Spoken Language Translation (IWSLT 2015)*, Da Nang, Vietnam.

Cho, E., Niehues, J., and Waibel, A. (2017). NMT-Based Segmentation and Punctuation Insertion for Real-Time Spoken Language Translation. *Proc. Interspeech 2017*, pages 2645–2649.

Fügen, C., Waibel, A., and Kolss, M. (2008). Simultaneous translation of lectures and speeches. *Springer Netherlands, Machine Translation, MTSN 2008, Springer, Netherland*, 21(4), 22. November.

Ha, T.-L., Niehues, J., and Waibel, A. (2016). Toward multilingual neural machine translation with universal encoder and decoder. *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*.

Ha, T.-L., Niehues, J., and Waibel, A. (2017). Effective Strategies in Zero-Shot Neural Machine Translation. *Proceedings of the 14th International Workshop on Spoken Language Translation (IWSLT 2017)*.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2017). Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT Summit 2005*.

Macháček, D., Kratochvíl, J., Vojtěchová, T., and Bojar, O. (2019). A speech test set of practice business presentations with additional relevant texts. In *Statistical Language and Speech Processing*, pages 151–161, Cham, Switzerland. Springer Nature Switzerland AG.

Nguyen, T.-S., Stueker, S., Niehues, J., and Waibel, A. (2019). Improving sequence-to-sequence speech recognition training with on-the-fly data augmentation. *arXiv preprint arXiv:1910.13296*.

Niehues, J., Pham, N.-Q., Ha, T.-L., Sperber, M., and Waibel, A. (2018). Low-latency neural speech translation. In *Interspeech 2018*, Hyderabad, India, Sept. 2 - 6.

Niehues, J., Cattoni, R., Stüker, S., Negri, M., Turchi, M., Ha, T., Salesky, E., Sanabria, R., Barrault, L., Specia, L., and Federico, M. (2019). The iwslt 2019 evaluation campaign. In *Proceedings of the 16th International Workshop on Spoken Language Translation (IWSLT 2019)*, Hong Kong, November.

Pham, N.-Q., Sperber, M., Salesky, E., Ha, T.-L., Niehues,

J., and Waibel, A. (2017). KIT's Multilingual Neural Machine Translation systems for IWSLT 2017. *Proceedings of the 14th International Workshop on Spoken Language Translation (IWSLT 2017)*.

Pham, N.-Q., Nguyen, T.-S., Niehues, J., Müller, M., and Waibel, A. (2019). Very deep self-attention networks for end-to-end speech recognition. *Proc. Interspeech 2019*, pages 66–70.

Popel, M. and Bojar, O. (2018). Training Tips for the Transformer Model. *The Prague Bulletin of Mathematical Linguistics*, 110(1):43–70.

Popel, M., Macháček, D., Auersperger, M., Bojar, O., and Pecina, P. (2019). English-Czech Systems in WMT19: Document-Level Transformer. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 342–348, Florence, Italy, August. Association for Computational Linguistics.

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tufiş, D., and Varga, D. (2006). The jrc-acquis: A multilingual aligned parallel corpus with 20+ languages. In *In Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC*, pages 2142–2147.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In I. Guyon, et al., editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.