The COMPRISE Cloud Platform

Raivis Skadiņš, Askars Salimbajevs
Tilde, Vienibas gatve 75a, Riga, Latvia, LV-1004
University of Latvia, Raina bulvaris 19, Riga, Latvia, LV-1586
{raivis.skadins, askars.salimbajevs}@tilde.lv

Abstract

This paper presents the COMPRISE cloud platform that is developed in the H2020 project. We present an overview of the COMPRISE project, its main goals, components, and how the cloud platform fits in the context of the overall project. The COMPRISE cloud platform is presented in more detail – main users, use scenarios, functions, implementation details, and how it will be used by both COMPRISE's targeted audience and the broader language-technology community.

Keywords: cloud platform, voice dialog systems

1. Introduction

The COMPRISE project[1] (Cost-effective, Multilingual, Privacy-driven voice-enabled Services) is a Research and Innovation Action funded by the European Union's Horizon 2020 programme. It aims to develop the next generation of voice interaction technology that will be more affordable, inclusive and, above all, secure.

Voice-operated technologies and tools have multiplied in recent years, voice is rapidly replacing touch or text as the main means of interaction with modern devices. COMPRISE aims to support, this expansion by providing the tools and methodology to make voice interaction more secure, more cost-effective, and more inclusive for a variety of languages.

Due to the cost of voice data collection and labelling, current voice interaction technologies have a strong bias in favour of languages with a wider user base (such as English), thus potentially excluding some users. In addition, they often rely on cloud-based algorithms to analyse voice signals, but there are few guarantees (if any) regarding how data stored in the cloud is used and will be used in the future by cloud service providers. COMPRISE is employing deep learning methodologies to improve speech-to-text and machine understanding of different languages and domains. In addition, it aims to create a methodology that protects the users' data, in order to ensure their privacy.

2. Approach

COMPRISE implements a fully private-by-design methodology and tools to reduce the cost and increase the inclusiveness of voice interaction technologies. To do so, we focus on the following key technologies:

• privacy-driven transformations to delete private information from the users' speech and the corresponding text data obtained by speech-to-text (Srivastava et al., 2020; Quian et al., 2018; Sundermann and Ney, 2003; Chou et al., 2019),

• joint centralized (H2020 COMPRISE project, 2019) and local learning to train large-scale systems from these transformed data while personalizing them for every user in a privacy-preserving way,

• weakly supervised learning to leverage both multiple automatic labelers for all utterances and manual labeling for a few utterances thereby drastically reducing the human labeling cost (Tam et al., 2014; Byambakhishig et al., 2014; Oualil et al., 2015; Kang et al. 2014; Zhou et al., 2019),

• robust integration of machine translation (MT) with speech and dialog processing tools to translate on-the-fly from one language to another and to generate additional training data by translating data available in other languages.

Building on scientific advances, we are implementing a cross-platform software development kit (SDK) and a sustainable cloud-based platform (Figure 1).

The SDK and the platform will ease the design of multilingual voice-enabled applications and their advancement.

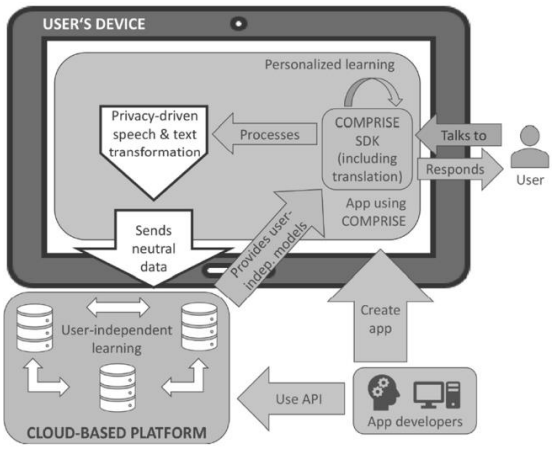

Figure 1. COMPRISE framework.

The COMPRISE framework leverages new web technologies largely supported by mobile browsers to make its solution also available for mobile devices, which are our primary targeted environment. The COMPRISE framework will not only provide a speech-to-text framework but rather a complete interactive conversational multilingual framework. The COMPRISE framework will embed the technologies in charge of analyzing, understanding and interpreting the voice of the user

[1] https://www.compriseh2020.eu

Proceedings of the 1st International Workshop on Language Technology Platforms (IWLTP 2020), pages 108–111
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

considering the spoken language, the accent, the mic encoding quality, etc.

the cost for both industrial providers and industrial users of voice interaction technologies.

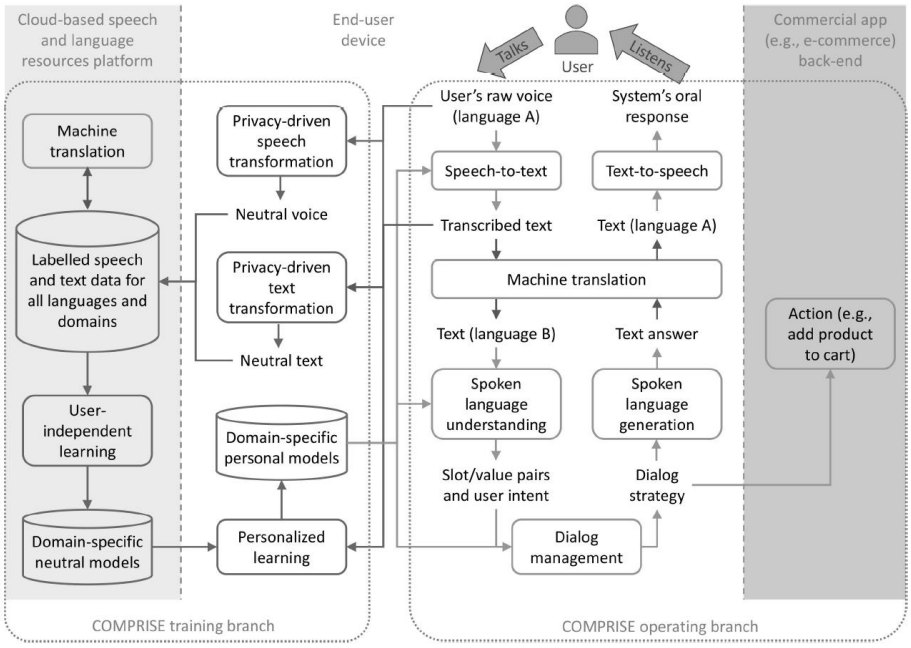

Figure 2. Detailed data flow of the COMPRISE voice interaction system.

The COMPRISE voice interaction system involves two branches running in parallel: the operating branch and the training branch (Figure 2).

The COMPRISE operating branch shown in the right half of the figure involves the usual chain of speech and language processing tools: speech-to-text, spoken language understanding, dialog management, spoken language generation, and text-to-speech. This branch conducts voice based interaction with the user in order to understand his/her request and fulfil it. This branch is similar to today's voice interaction systems, except that it runs locally on the user's device, it uses personalized models of speech and dialog, and it leverages machine translation to interface tools in different languages. The Operating branch is implemented as a cross-platform COMPRISE SDK that provides an easy to-use interface for multilingual voice-enabled application developers. By providing access to all tools developed within COMPRISE and by abstracting language- and platform-specific issues, the SDK significantly reduces the development time compared to existing SDKs from third-party vendors, thereby resulting in quicker time-to-market and major cost savings for industrial users of language technologies.

The COMPRISE training branch shown in the left half of the figure aims to collect large-scale in domain speech and language data for many languages and application domains and learn domain-specific personalized models from these data for speech-to-text, spoken language understanding, and dialog management in a privacy-preserving way. This branch is completely new and relies on research advances made in the project. This guaranties privacy and reduces

The COMPRISE Cloud platform is accessed by the SDK via REST API to exchange data and models. The platform will be used to store the neutral data and the models in a secure way, curate and label them, and update the models whenever sufficient additional data has been received. This platform fills a gap in the current ecosystem: existing resource repositories are good for speech resource description, dissemination, sharing, and distribution, but according to our knowledge there is no platform that would facilitate speech data creation, labelling, and curation. The COMPRISE platform is designed and developed for this purpose. It is a backbone on which all other components of the COMPRISE training branch are relying.

3. Cloud Platform

The neutral data and the corresponding (manual or automatic) labels are stored in the COMPRISE Cloud platform. The platform allows users to upload, store and manage data and labels and train or access large-scale user-independent models trained on these data. The platform functionality includes secure cloud-based data and model storage, scalable and dynamic cloud-based high-performance computing, APIs for continuous data upload and occasional model download, and general platform features (user interface, authentication, usage analytics, etc.) and procedures for data labeling and curation.

Two types of data will be handled by the platform: (1) speech and (2) text. The platform will allow training acoustic and language models for speech-to-text (STT), and intent detection models for spoken language understanding (SLU) on collected data. In the future support for other types of data and models might be added.

The main user of the cloud-based COMPRISE platform is a developer who uses COMPRISE SDK which will exchange data and models via REST API. Communications between the platform and the users' devices will be secured via state-of-the-art encryption and full compliance with the GDPR (e.g., regarding data retention) will be ensured.

3.1 User profiles

To specify user requirements for the COMPRISE Cloud platform first it's necessary to understand who platform users will be and how they will use the platform. We have identified four main user profiles – (1) COMPRISE Client apps, (2) developers, (3) data annotators and (4) administrators.

COMPRISE Client apps are machine users - applications that use COMPRISE SDK client components for STT and/or SLU and for communication with the COMPRISE Cloud platform. To achieve the best possible user experience, COMPRISE Client App wants to use the best neutral STT and/or SLU models for a particular usage domain. This is achieved by periodically (at runtime): (1) uploading new neutral speech and/or text data to the COMPRISE Cloud platform, and (2) downloading the latest models from the platform (e.g. on application start).

Developers use COMPRISE SDK to create voice-enabled privacy-preserving applications (e.g. personal assistant). To achieve the best possible user experience, the developer wants to use domain-specific STT and/or SLU models for the particular usage domain of the application. Developers use the COMPRISE Cloud platform to manage collected domain-specific neutral data, process collected data (e.g. apply machine translation, launch annotation tasks) and train domain-specific neutral STT and/or SLU models. After successful training, models are downloaded and used in developed applications. The collected data for each application are grouped into separate corpora, speech data is appended to the application speech corpus, text data is appended to the application text corpus. As collected data needs to be annotated, the developer shall be able to give access to the collected corpora to the other users - annotators.

Data annotator uses the COMPRISE Cloud platform to label domain-specific neutral speech and text data. Data annotators are granted access to speech or text corpora by Developers. Data annotators can have access to multiple corpora simultaneously. For speech corpora, annotators will provide transcription for each audio recording, but each user prompt in text data - intent label or next dialog state label. Labeled data is then used for the training of domain-specific neutral models.

The administrator maintains the COMPRISE Cloud platform and manages global access to its resources by creating, approving and deleting user accounts.

3.2 Architecture

The COMPRISE platform is expected to work in a cloud environment as several web-services using Containerization (e.g. Docker[2], Kubernetes[3]) technique. Therefore, hardware and system software management will be greatly simplified.

As seen in Figure 3 the COMPRISE platform consists of five main services:

• Authentication service authenticates users using a standard OpenID Connect protocol. As there are a lot of high-quality existing authentication solutions and providers, a new solution is not implemented in the scope of the project. Instead, existing authentication solutions or service providers (like Azure B2C) are utilized.

• API service provides COMPRISE platform functionality through an API. The service is implemented in the scope of the project.

• Storage service provides object storage through web service. Existing cloud storage solutions like Amazon S3 will be utilized as they provide scalability, high availability, low latency, durability and does not require hardware administration.

• Training service provides training of STT and SLU models. The service will be implemented in the scope of the project and use model training modules that are developed in the project's research activities. For machine translation of the training data, the service will use external machine translation service Tilde MT.

• Web UI provides a user interface for general COMPRISE platforms functions like registering applications, corpus annotation, triggering model training, etc.

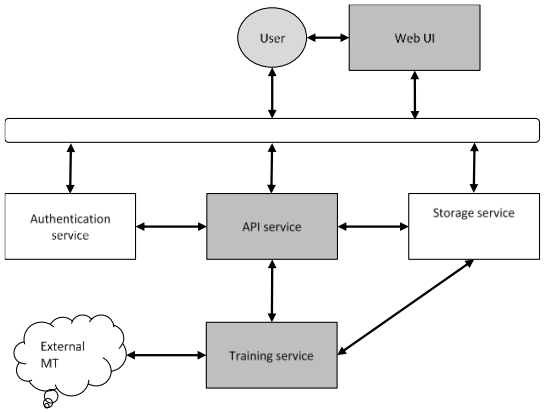

Figure 3. COMPRISE platform services

In order to efficiently balance the load between services and avoid unnecessary resource consumption COMPRISE platform API clients will request a special upload URL from API service, which allows uploading data to the Storage service directly without API service acting as an intermediary.

All services will be deployable as Docker containers which will allow to run them on almost any cloud provider infrastructure. For Docker container orchestration we use Kubernetes.

[2] https://www.docker.com/

[3] https://kubernetes.io/

Kubernetes scheduler and Horizontal Pod Autoscaler (HPA) are used to run containers only when they are requested and scale to multiple replicas when needed.

An optional gateway or proxy can be used for load-balancing, network administration, and protection.

3.3 Model Training

The training service is responsible for the training of in-domain neutral models for STT and SLU using model training modules provided by COMPRISE partners. These modules are packaged as Docker containers.

The training service is not exposed to the outside and is available only inside the cluster. It receives training requests from API service and initiates model training by starting training containers as a Kubernetes job. Started containers have direct access to the training data and models in the Storage service. Such an approach allows to run very different training workloads, improves portability and simplifies dependency maintenance (dependencies and environment are maintained inside containers). The limitation is that it does not allow to do traditional distributed training on multiple machines. We plan that in future this limitation can be lifted by using one or both of the following solutions:

• Model training containers can call Training service API to initiate sub-tasks.

• Training service can be extended to submit jobs to a classic High-Performance Cluster (HPC).

Also, in the future submission of training jobs to an external entity like the European Language Grid[4] will be considered. For machine translation of the training data, the service uses external machine translation service Tilde MT. In the future, support for other MT providers can be integrated.

3.4 User interface

The Web-based UI will provide the user interface for general COMPRISE Cloud platforms functions.

It is implemented using the Angular web framework and packaged as a Docker container as other services. It can be run directly in the cloud without an explicit virtual machine using services like Azure AppService or in the same Kubernetes cluster as other COMPRISE Cloud platform services.

The Web-based UI allows developers to sign-up, register applications, access API documentation and try-out forms. An important feature of the web-based UI is an interface for speech and text data annotation. This interface will be available without creating user accounts using a special URL with an embedded Annotator key, which will be created by the Developer and shared with annotators.

4. Development Status

The development of the platform started in November 2019 and is scheduled to be completed and made publicly available in August 2020. The platform architecture and API have already been specified and the first version of the API is already available in the test environment so that it can be integrated with the COMPRISE SDK.

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

5. Acknowledgements

COMPRISE has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 825081.

6. Bibliographical References

Byambakhishig, E., Tanaka, K., Aihara, R., Nakashika, T., Takiguchi, T., & Ariki, Y. (2014). Error correction of automatic speech recognition based on normalized web distance. In Fifteenth Annual Conference of the International Speech Communication Association (INTERSPEECH) (pp.2852–2856).

Chou, J. C., Yeh, C. C., & Lee, H. Y. (2019). One-shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization. In Proc. INTERSPEECH, (pp. 664–668)

H2020 COMPRISE project (2019). Deliverable D2.1: Baseline speech and text transformation and model learning library, Version 1.0, Retrieved from: https://www.compriseh2020.eu/files/2019/08/D2.1.pdf

Kang, S., Kim, J. H., & Seo, J. (2014). Post-error correction in automatic speech recognition using discourse information. Advances in Electrical and Computer Engineering, 14(2), 53-57.

Oualil, Y., Schulder, M., Helmke, H., Schmidt, A., & Klakow, D. (2015). Real-time integration of dynamic context information for improving automatic speech recognition. In Sixteenth Annual Conference of the International Speech Communication Association (INTERSPEECH), (pp 2107-21111).

Qian, J., Du, H., Hou, J., Chen, L., Jung, T., & Li, X. Y. (2018, November). Hidebehind: Enjoy Voice Input with Voiceprint Unclonability and Anonymity. In Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems (pp. 82-94).

Srivastava, B. M. L., Vauquier, N., Sahidullah, M., Bellet, A., Tommasi, M., & Vincent, E. (2020). "Evaluating voice conversion-based privacy protection against informed attackers", in Proc. ICASSP.

Sundermann, D., & Ney, H. (2003, December). VTLN-based voice conversion. In Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (IEEE Cat. No. 03EX795) (pp. 556-559). IEEE.

Tam, Y. C., Lei, Y., Zheng, J., & Wang, W. (2014, May). ASR error detection using recurrent neural network language model and complementary ASR. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2312-2316). IEEE.

Zhou, Z., Song, X., Botros, R., & Zhao, L. (2019, May). A Neural Network Based Ranking Framework to Improve ASR with NLU Related Knowledge Deployed. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6450-6454). IEEE.