Cost effective, Multilingual, Privacy-driven voice-enabled Services

www.compriseh2020.eu

Call: H2020-ICT-2018-2020
Topic: ICT-29-2018
Type of action: RIA
Grant agreement N°: 825081

WP N°: 2	**Privacy-driven voice interaction**
Deliverable N° D2.1	**Baseline speech and text transformation and model learning library**
Lead partner	**INRIA**
Version N°	**1.0**
Date	**30/08/2019**

Document information	
Deliverable N° and title	**D2.1– Baseline speech and text transformation and model learning library**
Version N°	**1.0**
Lead beneficiary	**INRIA**
Author(s)	**David Adelani (USAAR), Thomas Kleinbauer (USAAR), Imran Sheik (INRIA), Brij Srivastava (INRIA), Marc Tommasi (INRIA), Nathalie Vauquier (INRIA)**
Reviewers	**Dietrich Klakow (USAAR), Gerrit Klasen (ASCO)**
Submission date	**30/08/2019**
Due date	**31/08/2019**
Type[1]	**OTHER**
Dissemination level[2]	**PU**

Document history			
Date	**Version**	**Author(s)**	**Comments**
09/08/2019	0.1	David Adelani, Thomas Kleinbauer, Imran Sheik, Brij Srivastava, Marc Tommasi, Nathalie Vauquier	Draft deliverable
16/08/2019	0.2	David Adelani, Thomas Kleinbauer, Imran Sheik, Brij Srivastava, Marc Tommasi, Nathalie Vauquier	Revision based on the reviewer comments
30/08/2019	1.0	Emmanuel Vincent	Final version reviewed by the coordinator

[1] **R**: Report, **DEC:** Websites, patent filling, videos; **DEM:** Demonstrator, pilot, prototype; **ORDP:** Open Research Data Pilot; **ETHICS:** Ethics requirement. **OTHER:** Software Tools

[2] **PU:** Public; **CO:** Confidential, only for members of the consortium (including the Commission Services)

Document summary

This deliverable is devoted to the design, implementation, and evaluation of baseline transformations focusing on deleting the citizen's[3] identity and words carrying critical information, and model learning. It consists of this report and a list of software components that are available to the public on the COMPRISE git repository[4]. These software components are the basis of the future application programming interface (API) of COMPRISE. The report first recalls in Section 2 the parts of the COMPRISE architecture dedicated to speech and text transformations and provides a short description of the software components that are included in the deliverable. More details are provided directly in the software repository. The study of several scientific approaches behind speech and text transformations is presented in Section 3. This section is based on selected publications of the consortium and on extensive analysis and use of state-of-the-art methods. Experimental results are reported in Section 4. The complete loop that chains speech and text transformations followed by the composition of new datasets containing neutral information and then learning new automatic speech recognition tools is evaluated in Section 5.

[3] In this report we will use the terms "user" and "citizen" to refer to the person speaking to the dialogue system. From the GDPR point of view, the citizen is the "data subject".

[4] `https://gitlab.inria.fr/comprise`

Contents

1 Introduction

Modern applications now allow the user's voice to be the main interaction with computers or smart objects. The COMPRISE framework will embed the technologies in charge of analysing, understanding and interpreting the voice of the user taking into account the spoken language, the accent, the mic encoding quality, etc. Many of these technologies rely on training machine learning models from user's data, mainly the voice in our case. This raises serious privacy concerns because voice is considered as a biometric data and carries a lot of private and sensible information. One of the objectives of COMPRISE is to design the framework, that is the training and the use of the voice interaction tool, in a privacy-by-design manner. In COMPRISE, the voice interaction chain consists of two branches: the operating branch and the training branch (see Figure 1). Privacy in the operating branch will be ensured by running all computations[5] on the user's device and sending the final dialogue outcome only to the company delivering the desired service. This raises engineering challenges which are already solved to a great extent. Therefore, COMPRISE will focus on ensuring privacy in the training branch. To do so, we will introduce two innovations that complement each other: a new privacy-driven speech transformation and a new privacy-driven text transformation.

The privacy-driven speech transformation is applied to the citizen's speech signal before it is sent to the cloud in order to learn large-scale user-independent speech-to-text models from the speech data gathered from all users. The proposed transformation will result in a new "neutral" speech signal from which sensitive attributes related to the user's identity and to the traits and states of his/her voice have been removed, while keeping enough information to train a speech-to-text tool. Important challenges include how to learn such transformations of speech signals and how to do this with little training data from the user (sometimes a single short utterance). We have followed two approaches to build neutral speech representations. The first one relies on a generative adversarial network trained to reconstruct the original signal (this is required for human labelling purposes) and to favour correct phonetic decoding on the one hand and to favour incorrect speaker/traits/states classification on the other hand. The second one relies on a direct modification of the speech signal (envelope and pitch) inspired by speech conversion techniques, but with some randomisation that prevents the inversion to the original speech signal.

The privacy-driven text transformation operates on the outputs of the speech-to-text component, and identifies potentially privacy-threatening words, phrases, or expressions. We face three major challenges for text transformations: how to reliably detect privacy-threatening information in text and how to proceed with such information once it has been identified. The first point is challenging because of the open-ended nature of natural language. There is a great variety in the way the same information can be expressed and reliably detecting all of them cannot be expected from automatic means. Thus, it will be interesting to find means to quantify the privacy guarantees COMPRISE can provide. Also, what should be considered private information and what shouldn't is highly context-dependent which further increases this challenge. Thirdly , any robust transformation will have to face mistakes made by the speech-to-text models in order to minimise missing relevant words. Once privacy-threatening portions have been identified in an utterance made by the user, there are different ways how to conceal the private information, including deleting the words in question or replacing them with alternatives. However, we need to keep the requirements of subsequent dialogue processing in mind where changing the original user input too drastically might result in a severe degradation of the system performance. We will thus study the effect of several contending

[5] A possible exception could be made for certain components such as machine translation that could be run in a protected and trusted personal server. This point is still under discussion.

transformation strategies on typical NLU tasks.

Finally, any words detected as to be deleted by the privacy-driven text transformation will also be deleted in the corresponding speech signal by discarding the corresponding time intervals. We implement our approaches in a transformation library. We then report experiments on learning from neutral text and speech issued from our transformation library.

In the following we describe the proposed architecture of the software associated with this deliverable. The study of several scientific approaches is described in Section 3. We report scientific results and experiments for each approach in Section 4. Overall experiments for the complete chain following the proposed architecture are given in Section 5. Technical details are provided in Appendix A.

2 Architecture

2.1 Overall description and interactions

The global architecture is depicted in Figure 1. We have represented the main tasks and the flow of information between them.

The general objective is to chain a set of classical tools to provide voice services on the device: automatic speech recognition (ASR) a.k.a. speech-to-text, natural language understanding (NLU) and dialogue management (top right box of Figure 1). The chain is *called the operating branch* but one originality of COMPRISE is to improve the operating branch using machine learning techniques in a private-by-design manner. The *learning branch* has several components. A set of components are run on the device and are dedicated to the transformation of user data (speech and text) into neutral data (yellow column in the middle of Figure 1). Neutral data are sent to the cloud platform to be stored and exploited in off-line machine learning tools (gray-light blue box, bottom-right). The outcomes of the learning phase are new ASR modules or new NLU modules that can be personalised on the device (yellow box, middle right), first to replace original modules. To be able to transform original data into neutral data, two initial tasks are performed outside of the COMPRISE app and platform: from public datasets, we design speech and text transformation tools which will be part of the COMPRISE client app installed on the devices. The builders of the transformation tools are described in Section 3.

A full description of the API (builders and transformers) will be described in a forthcoming deliverable of work package (WP) 5.

2.2 Learning branch

In the learning branch, the aim is to produce neutral voice and text data that will be used for further improvement of the ASR, NLU and dialogue management components used in the COMPRISE system.

First the citizen voice is processed by an ASR module to obtain the corresponding text marked with temporal positions. Text is then processed by a *text transformer* that remove sensitive words and expressions. Speech signal is also processed by a *voice transformer* to produce a neutral voice. Thanks to the temporal positions and the neutral text, a *secure voice builder* is able to reconstruct a new speech signal where sensitive patterns have been removed.

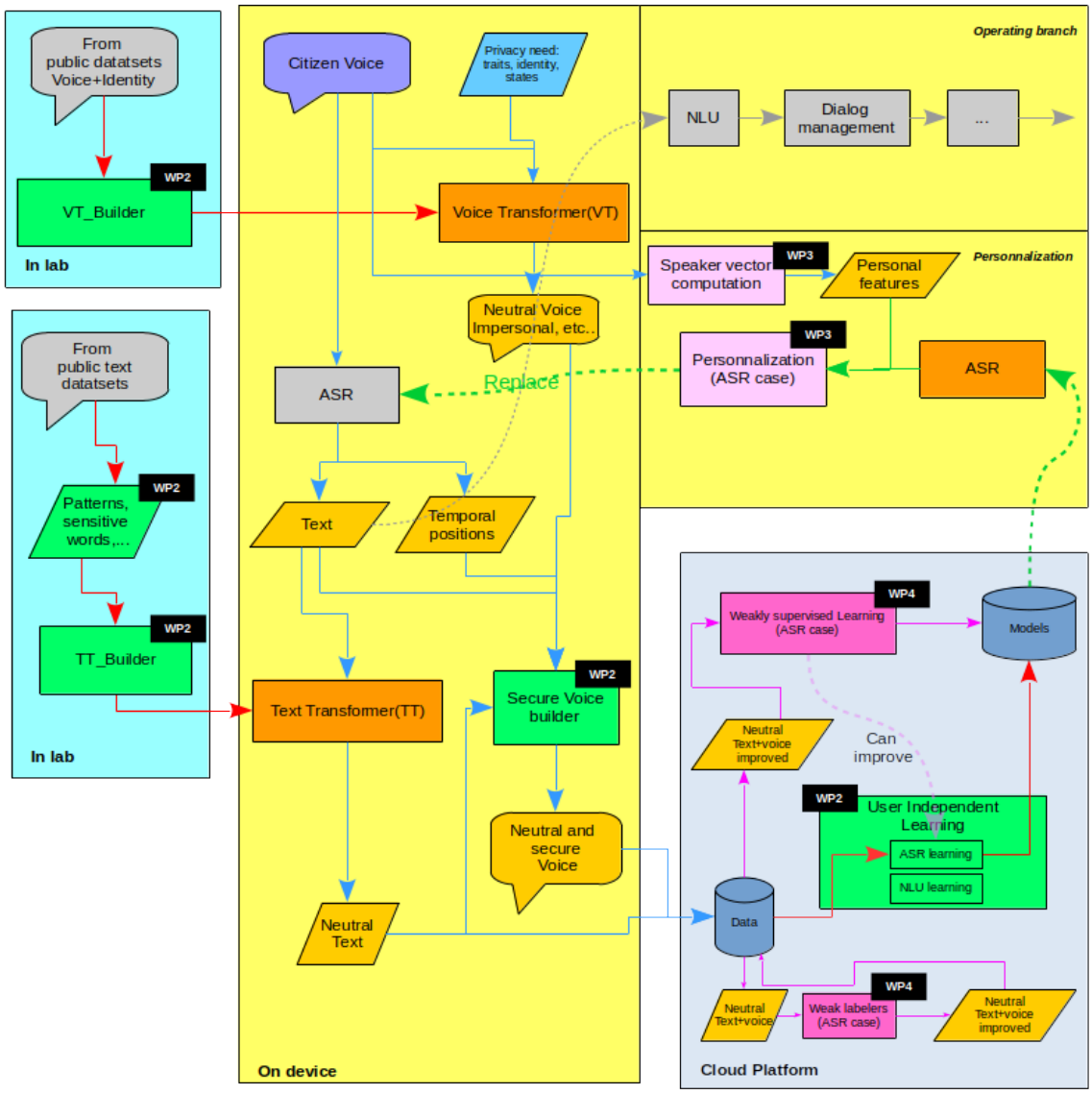

Figure 1: Overview of the global architecture of the learning branch, with interactions with other WPs.

`VoiceTransformer` objects have two main methods. The first one, `fit`, is used to compute internal parameters to fit the transformer to the user's voice. Indeed, some transformations need to be instantiated with specific features of the user's voice. It needs a few utterances as input and is used only once. Then `transform` performs a transformation to a target voice.

`TextTransformer` objects address two tasks: identifying the parts of the text to be transformed, and performing the actual transformation into a neutral text.

`SecureVoice_Builder` is included in the `word_masking_tool` package (see Appendix A.7). The `mask_words_in_speech` script in the `word_masking_tool` package receives neutral voice, neutral text and temporal positions of each sensitive word to reconstruct a secured speech signal where sensitive words have been removed.

The temporal position of each word in the speech signal is provided by the ASR module. However, for evaluation purposes, ground truth text can be used as input to the *text transformer* instead of the ASR text. In this case, the additional package `align-s2t` can be used to obtain the temporal positions of the words from the ground truth text in the speech signal.

The `mask_words_in_speech` script gives fine control over the type of words and named entities (NEs) that can be finally masked in the speech signal. Hence, the `word_masking_tool` package also includes an optional helper script `mask_words_in_text` which produces neutral text with the same type of words and NEs that are masked in the speech signal. Again, this feature is meant for detailed evaluation purposes.

Two specific functions are producing the text and the voice transformers. They are represented in the blue frame on the left of the diagram.

`VT_Builder` takes as input a (public) dataset containing speech data and speaker identities and produces a voice transformer. Two different classes of `VoiceTransformer` objects are produced for this preliminary version of the learning branch. Details about the transformations are given in Section 3.2.

`TT_Builder` is responsible for selecting the appropriate transformation strategy to use inside the `TextTransformer` and for continuously improving the models used by the latter. At the current stage, however, the `TextTransformer` uses fixed, external models only, pushing the need for a `TT_Builder` to a later point in the project. However, we have already experimented with different transformation strategies, as detailed in Section 3.3.

3 Scientific approaches

3.1 Privacy in voice technologies

Speech is a kind of biometric data that carries a lot of information. And of course, the text associated with speech and its meaning can be source of important privacy concerns. The GDPR frames the use of personal data but also defines the notion of personal data and some elements needed for a categorisation of personal data. A first effort done by the partners of the COMPRISE project, Álvaro Moretón's team at ROOT, was to propose such a categorisation in order to better identify sensitive words and patterns in texts, but also important features of speech that could be considered as sensitive.

The main source of inspiration was given by the The International Association of Privacy Professionals (IAPP) website.[6] Examples of other categorisations can also be extracted from different sources of legal documents (Recitals (14), (15),(26) and (30) and Articles 2, 4(1) and 9 of the GDPR, Data Protection Working Party: WP 01245/07/EN, WP 136 Opinion 4/2007 on the concept of personal data, see Appendix A.2). They do not provide a closed list. Some of the possible categories extracted from these document are the following:

- private and family life,
- working relations,
- types of activities that are undertaken by the individual,
- economic behaviour,
- social behaviour,
- physiological characteristics,
- living traits,
- appearance of the person,
- profession,
- group to which he/she belongs (age, occupation, place of residence).

The complete categorisation (including examples) is reported in Appendix A.1. It will be used to define the text transformation tools. For speech transformation, we have identified potential sources of privacy in the following features. Speaker identity in the first place, but also

- gender,
- age,
- ethnic origin,
- nativeness,
- personality,
- likeability,
- social status,
- sexual orientation,
- long-term disease (Parkinson, autism...).

A non-exhaustive list of states considered as private or sensitive information are

- health condition (cold...),

[6] https://iapp.org/resources/article/categories-of-personal-data/.

- intoxication,
- sincerity,
- deception,
- sleepiness,
- conflict,
- emotion (affect, crying),
- cognitive load.

These three lists pave the way for long-term research and innovation activities that cannot be handled in the first release of the COMPRISE project. For this release, we have considered a more focused goal, concentrating on speaker identity and some generic text processing. We are convinced that, based on the findings along the project duration, it will be possible to extend the list of handled private information in future generations of the COMPRISE platform.

3.2 Speech transformation

State-of-the-art speech processing algorithms can infer not only the spoken contents from the speech signal, but also the speaker's identity [Rey95], intention [Gu+17; HS16; BB13; Sto+98], gender [Zen+06; KK08], emotional state [EKK11; VK04; Kwo+03], pathological condition [DNB02; UK05; Sch+13], personality [SB13; Sch+15] and cultural [Sek97; VPB09] attributes to a great extent. These algorithms require just a few tens of hours of training data to achieve reasonable accuracy, which is easier than ever to collect via virtual assistants. The dissemination of voice signals in large data centres thereby poses severe privacy threats to the users in the long run.

These privacy issues have little been investigated so far. The most prominent studies use homomorphic encryption and bit string comparison [Pat12; Gla+17]. While these methods provide strong cryptographic guarantees, they come at a large computational overhead and can hardly be applied to state-of-the-art end-to-end deep neural network based systems.

Therefore, in COMPRISE we consider the following private information relating to the individual voice characteristics:

1. the user's identity
2. more general traits (gender, age, ethnic origin, nativeness, etc.) and states (health condition, intoxication, sincerity, etc.) which might be used to discriminate or impersonate him/her.

This task aims to transform the user's speech into a neutral speech signal from which this information can't be inferred anymore, while preserving the phonetic information required for human labelling and speech-to-text training.

3.2.1 Adversarial approach to hide user identity

In a first piece of work, we followed an approach based on adversarial training. The goal is to learn a representation of speech that performs well in ASR while hiding the speaker identity.

Several studies have recently exploited adversarial training for the goal of improving ASR performance by making the learned representations invariant to various conditions. While general acoustic variabilities have been studied [Ser+16], there is some work specifically on speaker invariance [Tsu+18; Men+18]. Interestingly, there is no general consensus on whether it is more appropriate to use speaker classification in an adversarial or a multi-task manner, despite the fact that these two strategies implement opposite means (i.e., encouraging representations to be speaker-invariant or speaker-specific). This question was studied in [Adi+18], in which the authors concluded that both approaches only provide minor improvements in terms of ASR performance. Their speaker classification experiments also show that the baseline system already tends to learn speaker-invariant features. However, they did not run speaker verification experiments and hence did not assess the suitability of these features for the goal of anonymisation.

In our work, we used the end-to-end ASR framework presented in [Wat+17] as our baseline. This architecture implemented in the ESPnet toolkit [Wat+18] is composed of three sub-networks: an *encoder* which transforms the input sequence of speech feature vectors into a new representation ϕ, and two *decoders* that predict the character sequence from ϕ. We assume that these networks have already been trained using data previously collected by the service provider (which may be public data, opt-in user data, etc).

The first decoder is based on connectionist temporal classification (CTC) and the second on an attention mechanism. As argued in [Wat+17], attention works well in most cases because it does not assume conditional independence between the output labels (unlike CTC). However, it is so flexible that it allows non-sequential alignments which are undesirable in the case of ASR. Hence, CTC acts as a regulariser to prune such misaligned hypotheses. We denote by θ_{e} the parameters of the encoder, and by θ_{c} and θ_{a} the parameters of the CTC and attention decoders, respectively. The model is trained in an end-to-end fashion by minimising an objective function $\mathcal{L}_{\mathrm{asr}}$ which is a combination of the losses $\mathcal{L}_c$ and $\mathcal{L}_a$ from both decoder branches:

$$\min_{\theta_{\mathrm{e}},\theta_{\mathrm{c}},\theta_{\mathrm{a}}} \mathcal{L}_{\mathrm{asr}}(\theta_{\mathrm{e}},\theta_{\mathrm{c}},\theta_{\mathrm{a}}) = \lambda\mathcal{L}_c(\theta_{\mathrm{e}},\theta_{\mathrm{c}}) + (1-\lambda)\mathcal{L}_a(\theta_{\mathrm{e}},\theta_{\mathrm{a}}),$$

with $\lambda \in [0,1]$ a trade-off parameter between the two decoders.

We now formally describe the form of the two losses $\mathcal{L}_c$ and $\mathcal{L}_a$. We denote each sample in the dataset as $S_i = (X_i, Y_i, z_i)$, where $X_i = \{x_1, ..., x_T\}$ is the sequence of T acoustic feature frames, $Y_i = \{y_1, ..., y_M\}$ is the sequence of M characters in the transcription, and z_i is the speaker label. In the case of CTC, several intermediate label sequences of length T are created by repeating characters and inserting a special *blank* label to mark character boundaries. Let $\Psi(Y_i)$ be the set of all such intermediate label sequences. The CTC loss $\mathcal{L}_c(\theta_{\mathrm{e}},\theta_{\mathrm{c}})$ is computed as $\mathcal{L}_c = -\ln P(Y_i|X_i;\theta_{\mathrm{e}},\theta_{\mathrm{c}})$ where $P(Y_i|X_i;\theta_{\mathrm{e}},\theta_{\mathrm{c}}) = \sum_{\psi\in\Psi(Y_i)} P(\psi|X_i;\theta_{\mathrm{e}},\theta_{\mathrm{c}})$. This sum is computed by assuming conditional independence over X_i, hence $P(\psi|X_i;\theta_{\mathrm{e}},\theta_{\mathrm{c}}) = \prod_{t=1}^{T} P(\psi_t|X_i;\theta_{\mathrm{e}},\theta_{\mathrm{c}}) \approx \prod_{t=1}^{T} P(\psi_t;\theta_{\mathrm{e}},\theta_{\mathrm{c}})$. The attention branch does not require an intermediate label representation and conditional independence is not assumed, hence the loss is simply computed as $\mathcal{L}_a(\theta_{\mathrm{e}},\theta_{\mathrm{a}}) = -\sum_{m\in M} \ln P(y_m|X_i, y_{1:m-1};\theta_{\mathrm{e}},\theta_{\mathrm{a}})$.

Speaker-adversarial model Inspired by [Feu+18], we propose to extend the above architecture with a *speaker-adversarial* branch so as to encourage the network to learn representations that are not only good at ASR but also hide speaker identity. This branch models an adversary which attempts to infer the speaker identity from the encoded representation ϕ. We denote by θ_s the parameters of the speaker-adversarial branch. Given the encoder parameters θ_{e}, the goal of the adversary is to find θ_s that minimises the loss $\mathcal{L}_{\mathrm{spk}}(\theta_{\mathrm{e}},\theta_s) = -\ln P(z_i|X_i;\theta_{\mathrm{e}},\theta_s)$. Our new model

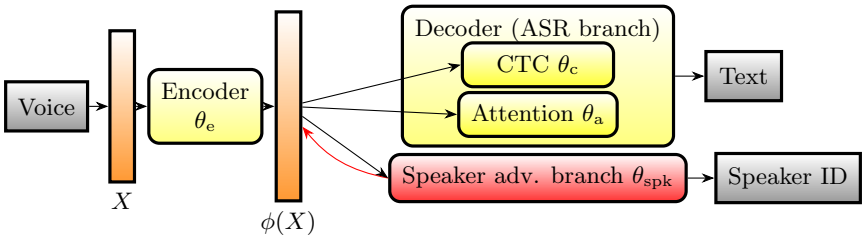

Figure 2: Architecture of the proposed adversarial approach. The speaker-adversarial branch is shown as a red box. The red arrow indicates *gradient reversal*.

is then trained in an end-to-end manner by optimising the following min-max objective:

$$\min_{\theta_e,\theta_c,\theta_a} \max_{\theta_s} \mathcal{L}_{asr}(\theta_e,\theta_c,\theta_a) - \alpha\mathcal{L}_{spk}(\theta_e,\theta_s),$$

where $\alpha \geq 0$ is a trade-off parameter between the ASR objective and the speaker-adversarial objective. The baseline network can be recovered by setting $\alpha = 0$. Note that the "max" part of the objective corresponds to the adversary, which controls only the speaker-adversarial parameters θ_s. The goal of the speaker-adversarial branch is to act as a "good adversary" and produce useful gradients to remove the speaker identity information from the encoded representation ϕ. In practice, we use a *gradient reversal layer* [Gan+16] between the encoder and the speaker-adversarial branch so that the whole network can be trained end-to-end via back-propagation. See Figure 2 for an illustration of the overall architecture.

3.2.2 Voice conversion

In the previous approach, the computed representation of speech is no more in the waveform domain. In order to be fully operational in the COMPRISE system, a speech waveform must be resynthesised from this representation. This resynthesis step is still subject to research and kept for future work. An alternative research direction is to operate a transformation on features that can easily be inverted back into a waveform so that the result is still audible and intelligible. Three approaches of the literature have been used. The first two are voice conversion approaches based on vocal tract length normalisation (VTLN) [CKA95; EG96]. After classical signal processing aiming to compute spectral envelope, pitch, and aperiodicity features, VTLN modifies the spectral envelope by frequency warping.

VoiceMask is described in [Qia+18; Qia+17]. The frequency warping is based on the composition of a quadratic function and a bilinear function using two different parameters. The inverse of this transformation is much more difficult to compute and therefore more resistant to attacks.

VTLN-Based-Conversion is described in [SN03]. Each speaker is represented by a set of spectra for k phonetic classes. The procedure maps a speaker to another one by finding the transformation parameters that minimise the distance between target class spectra and transformed source class spectra. k is an hyper-parameter of the method.

We have additionally introduced a modification of the pitch in both transformations. A third approach is based on neural networks. Recent advances in image translation from a domain to

another one now heavily rely on adversarial training (e.g. conditional-GAN). Extensions to multi-domain transformation have been proposed in [Cho+17]. In the speech literature, the same idea has been applied to perform voice conversion in [Kam+18]. It is therefore a natural extension of the VTLN-Based-Conversion of [SN03] using deep networks.

StarGAN-VC is described in [Kam+18]. We have used a publicly available implementation.[7]

We have performed experiments with these three methods.

3.3 Text transformation

What information is considered private can be a matter of preference and thus may differ from user to user. It also depends on the topic of conversation. For instance, mentioning an address might not be critical when two speakers discuss restaurant recommendations, but it could be more so when they talk about where they live.

For some initial experiments, we focus on negotiations of business meetings. Although this is not closely related to the COMPRISE use cases of WP6, we chose this domain as a starting point because of the availability of relevant dialogue data in the form of the VERBMOBIL corpus [Wah00] (see Section 5). In addition, business meetings are an appropriate domain since such conversations may contain multiple occurrences of potentially privacy-threatening information, including:

- **The identity of one or both speaker(s).** In the context of privacy, protecting the speaker's identity is a primary concern. However, it is not sufficient to just detect and hide mentions of names in the discourse since it may be possible to infer the speaker's identity through other means, for instance, by combining the dialogue information with external data sources ("linkage attacks"). This is a real risk, in particular when personal information, such as the speaker's age or gender [BGZ11; Sch+06; KAS02] or certain personality traits [Rab+17], can be inferred automatically from the utterances.

- **The company a speaker is associated with, and other relevant organisations.** If the identity of the speaker is not known, information about their professional background, such as their employer, may prove useful for linkage attacks. If the identity is already known, the same information could be abused for further profiling the speaker. In either case, it is often in the speaker's interest to conceal any organisations they are associated with. This becomes especially apparent when the organisations in question are, for instance, political parties or controversial entities.

- **The location of the meeting.** Information about locations discussed in the conversation may be suitable for further profiling the speakers.

- **The date and time of the meeting.** This information may become privacy-critical when combined with knowledge about the meeting location. Together, they allow an attacker to determine where the speaker will be at a given time. Multiple such data points lend themselves to the creation of movement profiles of the speaker. These bear the same potential for abuse as other forms of profiling (e.g., surveillance, unsolicited advertising, etc.). Knowledge about where the speaker will *not* be, e.g., not at home, leads to another level of threat when obtained by burglars.

[7] `https://github.com/liusongxiang/StarGAN-Voice-Conversion`

Instances of the above in a conversation may not directly threaten a speaker's privacy but especially in combination with external knowledge they may lead to insights that do. Even if no exact inferences can be drawn, information like the above could change an attacker's belief in how likely certain aspects about a speaker are to be true.

Once appropriate dialogue data has been collected in the course of COMPRISE, we intend to transfer our approach to the respective domains.

3.3.1 Detecting privacy-threatening information

The privacy classes listed in the previous section coincide with typical classes of named entity recognition (NER) tasks. We therefore consider an annotation scheme for the VERBMOBIL corpus in which sub-spans of utterances are annotated with one of the following labels: PER (Personal names), ORG (organisation), LOC (location), DATE and TIME. We do not identify demographic attributes such as gender, age and ethnicity of individuals participating in the dialogue, or any other potentially private information.

The VERBMOBIL corpus does not come pre-annotated with the above NE classes. For 80% of the corpus, we use the automatic NER tool `spaCy`[8] to identify the words or phrases in the transcripts that are subsequently considered as private information and manually correct the prediction mistakes. `spaCy`'s predictions are not perfectly reliable on a new dataset despite the fact that it was trained on a large and diverse corpus. We chose `spaCy` because it has a high NER accuracy of 85.85%[9] on the OntoNotes corpus [Hov+06] — a large annotated corpus comprising various genres of text (news, conversational telephone speech, weblogs, usenet newsgroups, broadcast, talk shows). The remaining 20% of the VERBMOBIL corpus was annotated by crowd-sourcing on the Figure-eight plaform.[10] We removed 24 dialogues with low quality or low agreement among crowd workers.

An example of an annotated dialogue using `spaCy` is show in Figure 3.

A: good morning , Misses Smith PER speaking , I am calling of course to make a date for the CEBIT ORG fair in Hanover LOC , ach Gott PER , in a month , March DATE of ninety five .
B: Mister Miller PER , I think for CEBIT ORG fair in Hanover LOC I would say March DATE the sixth , or we can say , Friday DATE , the tenth ?
A: I am sorry , on Friday DATE I don't have any time for it , I have another date , but what is about Monday DATE , the thirteenth ?
B: I think , it is not good , I would say , Tuesday DATE , on the fourteenth ?
A: I think , Tuesday DATE is okay , let us make a date for that day DATE .
B: yes, okay .
A: and now at a a congress in London LOC , and I would think , we get there by plane , one weekend DATE in April DATE ?
. . .

Figure 3: Excerpt from the beginning of a conversation in the VERBMOBIL corpus. Both A and B are native German speakers speaking English.

[8] `https://spacy.io`
[9] `https://spacy.io/usage/facts-figures`
[10] https://www.figure-eight.com/

3.3.2 Transformation strategies

The most straightforward way to hide private information from a transcript is to simply remove it. However, this may lead to problems for further NLU processing as the result would in most cases make the utterance difficult to parse. Consider, for instance, the last suggestion made by speaker A in the transcript of Figure 3. Assuming that both "one weekend" and "April" are private time information worth protecting, simply removing the words in question would leave only the word "in". Arguably, the result of this transformation is hardly useful anymore.

Placeholder tokens. A slightly richer transformation that keeps the original sentence structure intact replaces occurrences of private words with a special PLACEHOLDER token. For instance, the above example then becomes: "PLACEHOLDER in PLACEHOLDER?".

This approach makes the places where private information is retracted explicit. However, using a single placeholder token for all replacements might still prove insufficient as many different original utterances would result in the same utterance after the transformation. For instance, "One day in Madrid?" would equally lead to the above result.

An alternative could thus be to use typed placeholders which basically amounts to using the NE labels as the replacement tokens: "DATE in DATE?".

Same-type replacements. A downside of the placeholder approach is that it marks explicitly where a text transformation has occurred. As long as the automatic detection of private information is not perfect, an attacker can quite easily find out potentially private information by looking at all the NE instances that have *not* been replaced by a placeholder.

An approach to avoid this is not to use specific placeholder tokens as replacements but to exchange critical words with different words of the *same type*. For instance, a mention of a month such as "April" could be replaced by a different month, selected randomly. This way, even in the cases when the identification of private information fails to detect a relevant word occurrence, an attacker cannot easily[11] distinguish whether the words in the transformed transcript are the result of an actual transformation or whether they are the words originally uttered by the speaker. A *same-type* replacement can thus be seen as a noise component.

Randomly selecting alternative words raises the questions of how to construct the set of possible alternatives and which probability distribution over that set should be used in the sampling process. One possibility for the set of alternatives is to use an external source containing sufficient surrogates for all relevant types of NEs, such as Wikidata[12]. However, this may introduce surrogates that seem very foreign to the conversational domain, for example replacing non-popular business peoples' names by popular names like Donald Trump. Instead, we opt for using the set of all expressions with the same label *in the corpus itself*. We select a sensitive word from the set with a probability proportional to its relative frequency of occurrence in the source. This process has the additional advantage of not changing the expected distribution of NEs in the anonymised training corpus

Multi-word expressions. Ideally, when applying same-type replacements, each mention of a given entity should always be replaced with the same alternative in order to maintain text coherence. Thus, a mapping of original expressions to replacements should be maintained. However, the task is not trivial, since entities might be referred to by different expressions. For example, all of the following items refer to the same person:

[11] Note, however, that grammatical specificities, such as wrong prepositions, could inadvertently give clues about where replacements have occurred unless they also are part of the transformation. In the context of spontaneously spoken dialogue, such artifacts can, however, also occur naturally, especially with non-native speakers.

[12] `https://www.wikidata.org/`

1. Donald Trump
2. Donald J. Trump
3. Mr. Trump
4. President Trump
5. POTUS.

One way to address this problem, albeit imperfectly, is to map each word of such multi-word expressions individually rather than replacing the expression as a whole. That is, rather than mapping "Donald Trump" to, say, "Roger Smith", we maintain a mapping of "Donald" to "Roger" and "Trump" to "Smith". This will not help with expression #5 in the above list but does ensure that expressions #1 through #4 are replaced consistently.

In our experiments, we examine whether replacing multi-word expressions as a whole or word-by-word has a relevant effect on the subsequent NLU task and COMPRISE in general.

4 Scientific results

4.1 Speech transformation

We investigate how much of a user's *identity* is encoded in the transformed speech signal or representation. To this end, we conduct closed- and open-set speaker recognition experiments. The *closed-set* experiment refers to a classification setting where all test speakers are known at training time. The results are evaluated in terms of classification accuracy (ACC). In contrast, the *open-set* experiment (a.k.a. speaker verification) aims to measure the capability of an attacker to discriminate between speakers in a more realistic setting where the test speakers are not known beforehand. The results are evaluated in terms of equal error rate (EER). We implement the attacker with the state-of-the-art x-vector speaker recognition technique [Sny+18].

We use the Librispeech corpus [Pan+15] for all the experiments. We use different subsets for ASR training, adversarial training, and speaker verification. More details on the Librispeech dataset can be found on `http://www.openslr.org/12/`. A full description of the architecture, settings and datasets are given in the paper [Sri+19].

4.1.1 Adversarial approach

We train our speaker-adversarial network for $\alpha \in \{0, 0.5, 2.0\}$, leading to three encoded representations $\phi_\alpha(X)$. Recall that $\alpha = 0$ corresponds to the baseline ASR system as it ignores the speaker-adversarial branch. Table 1 summarises the results.

The first column presents the ACC and EER metrics obtained with the input filerbank features, which are consistent with the numbers reported in the literature. As expected, speaker identification and verification can be addressed to very high accuracy on those features. Using the encoded representation $\phi_0(X)$ trained for ASR only already provides a significant privacy gain: the ACC is divided by 2 and the EER is multiplied by 4, which suggests that a reasonable amount of speaker information is removed during ASR training. Nevertheless, $\phi_0(X)$ still contains some speaker identity information.

More interestingly, our results clearly show that adversarial training drastically reduces the performance in speaker identification but not in verification. On the other hand, and counter-intuitive to the speaker-invariance claims by several previous studies, we observe that the verification performance actually improves after adversarial training. This exhibits a possible limitation in the generalisation of adversarial training to unseen speakers and hence establishes the need for further investigation. The reason for the disparity between classification and verification performance might be that the speaker-adversarial branch does not inherently perform verification and hence is not optimised for that task. It might also be attributed to the representation capacity of that branch, to the number of speakers presented during adversarial training, and/or to the exact range of α needed for generalisable anonymisation. These factors of variation open several venues for future experiments.

Table 1: ASR and speaker recognition results with different representations. WER (%) is reported on *test-clean* set, ACC (%) on *test-adv* set and EER (%) on *test-clean-trial*.

	Filterbank	ϕ_0	$\phi_{0.5}$	$\phi_{2.0}$
WER	–	10.9	12.5	12.5
ACC	93.1	46.3	6.4	2.5
EER Pooled	5.72	23.07	21.97	19.56
EER Male	3.34	19.38	18.26	16.26
EER Female	7.48	26.46	24.45	22.45

We also notice that the WER stays reasonably low and stabilises to the value of 12.5% after increasing α from 0.5 to 2. In particular, for $\alpha = 2$ the WER is just 1.6% absolute more than the baseline ($\alpha = 0$).

A complete analysis and description is presented in [Sri+19].

4.1.2 Voice conversion

In a first set of experiments, we consider three different attack scenarios and several variants of the three transformers. Let us start with the description of the variants. Basically, all transformation methods are parametrised. Additionally, when converting the voice of a source speaker to a target speaker, there are two options: either the target speaker is fixed (unique) or it is drawn randomly. This results in three conversion strategies:

Strategy 1 All parameters are fixed, for instance all utterances are mapped to a unique speaker.

Strategy 2 A specific set of parameters is associated with (all utterances of) a speaker.

Strategy 3 A specific set of parameters is associated with each utterance.

Strategy 3 implies more randomness, which could result in increased privacy. However some averaging-based attacks could take benefit of this kind of transformation. Therefore, in some cases we have kept the three strategies. Specifically, we experiment the three methods in the following way:

VoiceMask We sample the parameters of the spectral envelope transform and the pitch transform uniformly at random in a given interval and follow strategy 3.

VTLN-Based-Conversion We apply the same strategy for the pitch and we fix the number of phonetic classes to 8. In each experiment a set of at most 100 speakers is selected uniformly at random. In strategy 1, only one target speaker is selected. In strategy 2, for each source speaker we select a target speaker at random among the 100. In strategy 3, for each utterance we select a target speaker at random among the 100.

StarGAN The strategy is similar to VTLN-Based-Conversion except that the transformation model is learned from the whole set of speakers before selecting at most 100 speakers and applying the transformation. We only consider strategy 3.

We evaluate the transformers on the Librispeech dataset which contains 1,212 speakers. The training set contains either 100 or 360 hours of speech. The baseline speaker verification results when no voice transformation is applied are given in Table 2.

Table 2: Baseline EER (%) when no speaker transformation is applied.

subset	EER%
Pooled	4.31
Male	0.89
Female	6.75

We consider three attack scenarios. In all scenarios, we consider that the platform and the device are secure. An attacker cannot have access for instance to the original voice of COMPRISE users. Neutral voice and neutral texts are publicly available. The question raised here is: *is an attacker able to guess who said what in this public dataset?* An attacker performs the following steps that also mimic the speaker verification procedure: (i) he trains a model to build x-vectors on some dataset (the training set); (ii) he computes the x-vectors of the speakers under attack (the enrolment set); (iii) he tries to identify whether the utterances in the test set are spoken by the user under attack or by impostors. In our situation the test set is the publicly available set of transformed voices. Note that the training set for x-vectors contains utterances pronounced by users that are not in the test and enrolment sets. The training and enrolment sets are always original voices that an attacker may be able to transform if he/she knows or guesses the transformation. Note that training x-vectors on the publicly available dataset is not possible because user identities are unavailable.[13] The transformation itself can be learned (e.g., in the StarGAN case) using possibly another training set.

The three attack scenarios are as follows:

Scenario 1 The attacker does not know that the voice was transformed. He trains a model for x-vectors on some public original (non-transformed) speech dataset. He enrols new speakers with their original voice but the test set is from the transformed COMPRISE dataset. This case correspond to a very weak attacker. It can be briefly summarised as:

- Training on the original data
- Enrol with original data

[13] When the transformation to neutral voice is purely deterministic, an attacker could try to forge new user identities. To perform this task, he/she should guess that two utterances are from the same user. We didn't perform such a test. In COMPRISE, it could be a problem if the structure of dialogues is available.

- Test on transformed data.

The results are given in Table 3. We have selected only strategy 3 for VoiceMask and StarGAN but all the strategies were evaluated for VTLN-Based-Conversion. We observe that speaker verification achieves better results than simple random guessing (there are 100 users in the test set). The increased computational complexity of StarGAN does not translate into a real improvement.

Table 3: EER (%) when the attacker does not know that the voice was transformed (Scenario 1).

	VoiceMask	**VTLN-Based-Conversion**			**StarGAN**
Subset	Strategy 3	Strategy 1	Strategy 2	Strategy 3	Strategy 3
Pooled	28.69	24.27	30.99	27.38	28.89
Male	30.07	22.94	29.62	26.73	22.72
Female	21.90	20.62	27.01	26.28	31.75

Scenario 2 The attacker knows the voice transformation method and the parameters:

- Training on the transformed data
- Enrol with transformed data
- Test on transformed data.

In that case, the attacker knows exactly the transformation. The attacker proceeds in the following way: he applies the known transformation to the training and enrolment sets; he learns the x-vector representation with the transformed training set; he computes the x-vector representation for users in the enrolment set and performs the classification steps to guess whether the test speaker is an impostor or not.[14] The transformation is known to the attacker because it is distributed as part of the code embedded in the COMPRISE user's device. Note that some transformations include some random steps when a user is mapped to another one. In that case, the attacker proceeds in the same way by a random mapping. Hence the transformation made by the attacker is not guaranteed to be exactly the same as the one performed on the public COMPRISE dataset. However, we expect to obtain results similar to the state of the art of speaker verification. This is confirmed in Table 4.

Table 4: EER (%) when the attacker knows the voice transformation method and the parameters (Scenario 2).

	VoiceMask	**VTLN-Based-Conversion**			**StarGAN**
Subset	Strategy 3	Strategy 1	Strategy 2	Strategy 3	Strategy 3
Pooled	5.01	4.71	3.91	6.32	7.32
Male	2.00	1.11	1.11	3.41	3.34
Female	7.66	7.30	5.47	8.94	10.22

Scenario 3 The attacker knows the voice transformation method but not the parameters:

[14] The computation of EER metrics is done using an oracle and is not part of the attack.

- Training on the transformed data
- Enrol with transformed data
- Test on transformed data.

In that case, the attacker knows the transformation up to the parameter values. Hence, the enrolment and the train sets are transformed but with a possible shift of the parameter values. The evaluation of this scenario is still pending.

Note that, in our experiments with StarGAN, we have used all speakers from the three sets to build the model. This makes the attacker very strong because in principle he/she would not have access to the users' speech and identities in the test set. Further experiments will be done with StarGAN with a more realistic attacker model.

4.2 Text transformation

Masking out relevant information will help protecting the users' privacy. But with some original information lost, how useful is the transformed text for NLU model training? This is the main question we are interested in. In the following, we present the experiments we designed to gauge the degradation inflicted by the information loss.

4.2.1 Experimental setup

By our definition of "private" for text transformations, the parts of the dialogue transcript that are affected by a privacy-preserving transformation are NEs. Thus it seems intuitive to assess the suitability of the transformation by measuring the performance degradation on an NER task when applied to transformed text.

The task of NER has been well studied and state-of-the-art systems reach high prediction accuracies, e.g., 93.09% on the CoNNL dataset [ABV18]. As we are mostly interested in the *relative* performance drop on privacy-transformed data rather than in absolute accuracy, we use the BiLSTM-CRF model in [HXY15], which consists of a bidirectional LSTM (BiLSTM) model with a subsequent conditional random field (CRF) decoding layer and achieves a decent NER accuracy both on original data as well as on the output of different privacy-preserving transformations. The word features used in the chosen implementation[15] are word embeddings from Glove [PSM14], that are then passed into the BiLSTM-CRF model for sequence tagging.

[15] https://github.com/zalandoresearch/flair

Table 5: Number of tokens in the training set for each NE class.

NE class	Number of tokens
PER	656
LOC	1,222
ORG	594
DATE	14,813
TIME	12,261
O	129,026

The VERBMOBIL corpus consists of 726 English dialogues labelled as explained in Section 3.3.1. For training the NER model, we divide the corpus into 508 training, 72 development, and 146 test dialogues, each with a variable number of sentences (19,151 training, 2,846 development, and 5,230 test sentences). The number of tokens for each NE class in the training set is shown in Table 5. Tokens that do not belong to the PER, LOC, ORG, DATE and TIME classes are assigned the label O.

Figure 4 shows our experimental setup. For all comparison experiments, we first apply the respective privacy transformation on the VERBMOBIL training set before training another BiLSTM-CRF model. The hyper-parameters of the model are the same as for the baseline model: 300-dimensional embedding layer (from Glove), 256 hidden LSTM layer size in each direction, initial learning rate of 0.1 (halved when the loss on the development set does not improve), mini-batch size of 32, and maximum number of 50 epochs. We then compare the performance of the models on the test set in terms of the F1-score. This metric is more appropriate than the usual accuracy metric when the number of false positives (FP) and false negatives (FN) differs substantially. We use the same test set for all experiments without any transformations applied to it.

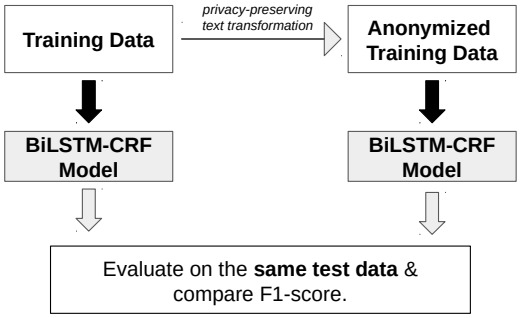

Figure 4: Experimental setup to measure the impact of privacy-preserving text transformation on a NER task. The various experiments differ in the type of transformation used to derive the anonymised training data.

4.2.2 Experimental results

The baseline to which we compare all other experiments is simply trained on the original training set, i.e., without removing any private information. On the test set, the resulting model yields an F1-score of 86.1%. We tried different variants of training data such as expressing all sentences in lower case and ignoring punctuation. This is motivated by the fact that the sentences output by ASR are often uncased and without punctuation. Table 6 shows the performance of the uncased (with/without punctuation) and cased NER models (with/without punctuation). There is no significant difference between the cased and uncased models, however the models ignoring punctuation are less accurate.

Table 7 shows the F1-score achieved for five different transformation strategies. For all strategies, we observe a similar drop in F1-score when we use different model variants (cased/uncased with or without punctuation). In the following, we focus on the performance of the cased model with punctuation (i.e., cased-punct). The first strategy which consists of simply replacing each word

Table 6: Macro F1-score average after pre-processing of the training data — lower casing (uncased) and ignoring punctuation (nopunct). Uncased-nopunct is the model trained on sentences that are in lower case and without punctuation, while cased-punct is the model trained on sentences with punctuation and the case of words in the sentences are intact.

Model	**Macro F1-score average**
cased-punct	86.1%
cased-nopunct	85.1%
uncased-punct	**86.3**%
uncased-nopunct	85.2%

Table 7: Macro F1-score average for five different transformation strategies. The one with asterisk is the transformation with the best accuracy.

Replacement Strategy	**cased, punct**	**cased, nopunct**	**uncased, punct**	**uncased, nopunct**
No Replacement	**86.1%**	**85.1%**	**86.3%**	**85.2%**
Replace All NEs with Placeholders	5.4%	3.2%	2.2%	3.0%
Replace All multi-word NEs with Placeholders	4.0%	0.3%	1.8%	4.0%
Replace All NEs with same-type NE	79.0%	78.9%	78.7%	79.1%
Replace All multi-word NEs with a single same-type NE	51.1%	49.0%	42.9%	53.1%
Replace All multi-word NEs with a multiword same-type NE*	84.5%*	84.1%*	83.5%*	83.3%*

labeled as a NE with the PLACEHOLDER token results in a drastic drop of the F1-score down to 5.4%. Replacing multi-word expressions with a single PLACEHOLDER token also does not help.

Replacing all words labeled as NEs with tokens of the same type yields a much smaller drop of 7% in F1-score. As shown in Table 8, we replace CEBIT by Royal, Hanover by Pittsburgh, Misses by Bob, Smith by Tanaka and "a month, March" by "Tuesday twenty eleventh Saturday". When replacing multi-word expressions by a single word, we see a larger drop down to 51.1% F1-score. Replacing multi-word expressions by other multi-word expressions (e.g., replacing "Misses Smith" by "Mark Lively") gives the best performance (84.5%), slightly lower than the baseline. This shows that the text obtained using same-type text transformations (with single/multi-word expression) can be efficiently used for training NLU systems while protecting the users' privacy.

Table 8: Example of the different replacement strategies to protect the identity of users and details of their appointments.

Replacement strategy	**Transformed text**
No Replacement	good morning , Misses Smith PER speaking , I am calling of course to make a date for the CEBIT ORG fair in Hanover LOC , ach Gott PER , in a month , March DATE .
Replace All NEs with Placeholders	good morning , PLACEHOLDER PLACEHOLDER PER speaking , I am calling of course to make a date for the PLACEHOLDER ORG fair in PLACEHOLDER LOC , ach PLACEHOLDER PER , in PLACEHOLDER PLACEHOLDER PLACEHOLDER PLACEHOLDER DATE .
Replace All multi-word NEs with Placeholders	good morning , PLACEHOLDER PER speaking , I am calling of course to make a date for the PLACEHOLDER ORG fair in PLACEHOLDER LOC , ach PLACEHOLDER PER , in PLACEHOLDER DATE .
Replace All NEs with same-type NE	good morning , Bob Tanaka PER speaking , I am calling of course to make a date for the Royal ORG fair in Pittsburgh LOC , ach Hof PER , in Tuesday twenty eleventh Saturday DATE .
Replace All multi-word NEs with a single same-type NE	good morning , Bob PER speaking , I am calling of course to make a date for the Royal ORG fair in Pittsburgh , ach Hof PER , in Tuesday DATE .
Replace All multi-word NEs with a multiword same-type	good morning , Mark Lively PER speaking , I am calling of course to make a date for the Royal ORG fair in Pittsburgh , ach Hof PER , in twenty seventh DATE .

4.3 Theoretical results on privacy-preserving learning

One of the concerns in COMPRISE is to obtain formal privacy guarantees for the proposed privacy-preserving voice interaction framework. Therefore, we also conducted more theoretical research on privacy-preserving learning.

4.3.1 Propagation and decentralisation

One of our research tracks concerns decentralized protocols. Indeed, one alternative to data centralization in a global server is to rely on peer-to-peer communications between the users. Data stays on the users' devices and the users just collaborate. One can expect better privacy guarantees and also a reduction of infrastructure costs. We want to evaluate whether this paradigm is suitable for large-scale machine learning tasks such as ASR for instance. A first step in that direction has been done in the paper [BGH19]. The approach relies on *gossip protocols* and the paper studies their impact on privacy.

Gossip protocols, also called rumor spreading or epidemic protocols, are widely used to disseminate information in massive peer-to-peer networks. These protocols are often claimed to guarantee privacy because of the uncertainty they introduce on the node that started the dissemination. But this claim has not been formally studied until now. Our paper is the first to study gossip protocols using a rigorous mathematical framework based on differential privacy to determine the extent to which the source of a gossip can be traceable. Considering the case of a complete graph in which a subset of the nodes are curious, we derive matching lower and upper bounds on differential privacy showing that some gossip protocols achieve strong privacy guarantees. Our results further reveal an interesting tension between privacy and dissemination speed: the standard "push" gossip protocol has very weak privacy guarantees, while the optimal guarantees are attained at the cost of a drastic increase in the spreading time. Yet, we show that it is possible to leverage the inherent randomness and partial observability of gossip protocols to achieve both fast dissemination speed and near-optimal privacy.

4.3.2 Computing privately on pairs of data

The problem of collecting aggregate statistics from a set of users is also at the center of the data processing that could be done in the context of COMPRISE. In a privacy-by-design approach, we consider that individual contributions belonging to some (discrete or continuous) domain must remain private even from the data analysts, or the aggregation platform.

In the popular local model of differential privacy, users apply a local randomizer to their private input before sending it to an untrusted aggregator. In this context, most work has focused on computing quantities that are separable across individual users, such as sums and histograms. In the paper [Bel+19], members of INRIA in collaboration with other researchers at EPFL study the novel problem of privately computing quantities that come in the form of averages over pairs of data points. This is known as the (degree 2) U-statistic. The class of U-statistics covers many statistical estimates of interest, including Gini mean difference, Kendall's tau coefficient, Wilcoxon Mann-Whitney hypothesis test and Area under the ROC Curve (AUC). These statistics are also commonly used as empirical risk measures for machine learning problems such as ranking, clustering and metric learning. Our contribution is the design and analysis of several protocols for private estimation of U-statistics in the local differential privacy model with privacy and utility guarantees.

5 Experiments on the complete learning branch

In order to evaluate the joint effect of speech and text transformations, we conducted experiments on the VERBMOBIL corpus. The VERBMOBIL project [Wah00] (1993–2000) was a research project supported by German research funds and companies that addressed the development of a speech-to-speech translation system for English, German and Japanese. In the course of the project, a large corpus[16] of spontaneous telephone conversations by a total of 54 speakers was collected and annotated [WRS02]. In each conversation, the two speakers were tasked with the negotation of a business meeting. This included agreeing on a date and time as well as a location for the meeting, and additional points of negotiation such as leisure time activities. The VERBMOBIL corpus is multilingual but we only consider the English subset in our experiments.

In the following, we apply both speech and text transformations on the VERBMOBIL corpus. We then train an ASR model on the transformed corpus and compare its performance with that of an ASR model trained on the original VERBMOBIL corpus.

5.1 ASR Setup

5.1.1 Training-development-test split

We split the VERBMOBIL corpus into three datasets for training, development, and test:

- All dialogues in CDs 47, 52, 55 and 56 (including *infodesk* domain + German English speakers) and dialogues q010-q020 from CD 6 (including German English speakers) are put into the test set. Dialogues from the other CDs including those speakers are also put into the test set.
- All dialogues in CDs 32 and 51 (including *infodesk* domain + German English speakers) and dialogues q001-q009, q021 and q022 from CD 6 (including German English speakers) which

[16] `https://www.phonetik.uni-muenchen.de/Bas/BasVM1eng.html`
`https://www.phonetik.uni-muenchen.de/Bas/BasVM2eng.html`

are not part of the test set are put into the development set. Dialogues from the other CDs including those speakers are also put into the development set. Moreover, to ensure balanced development and test WERs, the dialogues corresponding to speakers HCF, QZO, and QZB are moved from the test set to the development set.

- All the remaining dialogues are put into the training set, except for speakers ADB (British English speaker) and VNC (with Asian accented English) which are put into the test set and the development set, respectively.

5.1.2 ASR models

We train three different ASR acoustic models on the training set, namely a hidden Markov model with Gaussian mixture model output (GMM-HMM) based tri-phone model [Pov+11], a time delay neural network (TDNN) based model [PPK15], and the TDNN chain model [Pov+16]. The language model is a tri-gram model. The development set is mainly used to tune hyper-parameters such as the language model weight. The WER performance is reported on the test set.

5.2 Data transformation

To obtain the transformed VERBMOBIL corpus, the following steps are applied:

- ASR, with the best performing Chain acoustic models [Pov+16], is trained on the original VERBMOBIL English corpus.
- Text transformation is applied on the resulting ASR transcriptions of the train, development and test sets by removing sensitive words, which are words corresponding to PER (Personal names), ORG (organization), LOC (location), DATE and TIME.
- VTLN utterance based speech transformation is applied on the original VERBMOBIL English corpus speech data.
- Given the word level speech-text time alignments, the sensitive words are removed from the transformed speech utterances, by replacing these words with silence.

5.3 ASR performance results

Table 9 presents the WER performance of the ASR models trained on the original and transformed VERBMOBIL corpora. **tri3_si** and **tri3** denote GMM-HMM based tri-phone acoustic models with speaker-independent and speaker-adaptive decoding, while **nnet3** and **chain** represent the TDNN and TDNN Chain acoustic models, respectively. The absolute WER is larger than on Librispeech, due to the smaller size of the training set. However, we are more interested in the relative performance gap resulting from the data transformation. We observe that the transformation may hurt the global performance of the model, but its impact is limited in the case of the TDNN Chain model of [Pov+16]. The next steps will be to tune the trade-off between the ASR performance and the privacy guaranties.

Table 9: WER (%) achieved when training ASR on transformed speech and text data.

VERBMOBIL corpus	**tri3_si**	**tri3**	**nnet3**	**chain**
original	34.71	28.86	25.45	22.36
transformed	44.56	39.90	34.89	25.51

6 Conclusion

In this report, we have described our scientific advances on the tasks of removing identity from speech signals and removing sensitive information from texts. We have proposed several solutions, and investigated solutions available in the literature. We have also conducted an overall evaluation on a dialogue corpus (VERBMOBIL), which shows that training an ASR tool on the neutral data output by our speech and text transformation tools is effective. As a conclusion, we observe that the research topic of hiding private information from speech and text is still open and will need more effort to obtain acceptable performance in real-world scenarios.

The software tools used in all experiments reported in this document are detailed in Appendices A.3, A.4, A.5, A.6, and A.7 and have been made publicly available. These tools will form the basis for the future API of the learning branch in the COMPRISE architecture.

References

[ABV18] A. Akbik, D. Blythe, and R. Vollgraf. "Contextual String Embeddings for Sequence Labeling". In: *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 1638–1649.

[Adi+18] Y. Adi et al. "To Reverse the Gradient or Not: An Empirical Comparison of Adversarial and Multi-task Learning in Speech Recognition". In: *arXiv preprint arXiv:1812.03483* (2018).

[BB13] T. Ballmer and W. Brennstuhl. *Speech act classification: A study in the lexical analysis of English speech activity verbs*. Vol. 8. Springer Science & Business Media, 2013.

[Bel+19] A. Bellet et al. "Private Protocols for U -statistics in the Local Model and Beyond". In: *Privacy Preserving Machine Learning Workshop at the ACM Conference on Computer and Communications Security*. 2019.

[BGH19] A. Bellet, R. Guerraoui, and H. Hendrikx. "Who started this rumor? Quantifying the natural differential privacy guarantees of gossip protocols". In: *arXiv e-prints* (Feb. 2019).

[BGZ11] J. D. Burger, J. H. and George Kim, and G. Zarrella. "Discriminating gender on Twitter". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '11)*. Edinburgh, UK: Association for Computational Linguistics, July 2011, pp. 1301–1309.

[Cho+17] Y. Choi et al. "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation". In: *CoRR* abs/1711.09020 (2017).

[CKA95] J. Cohen, T. Kamm, and A. G. Andreou. "Vocal tract normalization in speech recognition: Compensating for systematic speaker variability". In: *The Journal of the Acoustical Society of America* 97.5 (1995), pp. 3246–3247.

[DNB02] A. A. Dibazar, S. Narayanan, and T. W. Berger. "Feature analysis for automatic detection of pathological speech". In: *2nd Joint EMBS-BMES Conference*. Vol. 1. 2002, pp. 182–183.

[EG96] E. Eide and H. Gish. "A Parametric Approach to Vocal Tract Length Normalization". In: *Proceedings of the Acoustics, Speech, and Signal Processing, 1996. On Conference Proceedings., 1996 IEEE International Conference - Volume 01*. ICASSP '96. Washington, DC, USA: IEEE Computer Society, 1996, pp. 346–348.

[EKK11] M. El Ayadi, M. S. Kamel, and F. Karray. "Survey on speech emotion recognition: Features, classification schemes, and databases". In: *Pattern Recognition* 44.3 (2011), pp. 572–587.

[Feu+18] C. Feutry et al. "Learning Anonymized Representations with Adversarial Neural Networks". In: *arXiv preprint arXiv:1802.09386* (2018).

[Gan+16] Y. Ganin et al. "Domain-adversarial training of neural networks". In: *JMLR* 17.1 (2016), pp. 2096–2030.

[Gla+17] C. Glackin et al. "Privacy Preserving Encrypted Phonetic Search of Speech Data". In: *2017 IEEE ICASSP*. 2017, pp. 6414–6418.

[Gu+17] Y. Gu et al. "Speech intention classification with multimodal deep learning". In: *Canadian Conference on Artificial Intelligence*. 2017, pp. 260–271.

[Hov+06] E. Hovy et al. "OntoNotes: The 90% Solution". In: *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*. NAACL-Short '06. 2006, pp. 57–60.

[HS16] N. Hellbernd and D. Sammler. "Prosody conveys speaker's intentions: Acoustic cues for speech act perception". In: *Journal of Memory and Language* 88 (2016), pp. 70–86.

[HXY15] Z. Huang, W. L. Xu, and K. Yu. "Bidirectional LSTM-CRF Models for Sequence Tagging". In: *ArXiv* abs/1508.01991 (2015).

[Kam+18] H. Kameoka et al. "StarGAN-VC: Non-parallel many-to-many voice conversion with star generative adversarial networks". In: *CoRR* abs/1806.02169 (2018).

[KAS02] M. Koppel, S. Argamon, and A. R. Shimoni. "Automatically Categorizing Written Texts by Author Gender". In: *Literary and Linguistic Computing* 17.4 (Nov. 2002), pp. 401–412.

[KK08] M. Kotti and C. Kotropoulos. "Gender classification in two emotional speech databases". In: *ICPR*. 2008, pp. 1–4.

[Kwo+03] O.-W. Kwon et al. "Emotion recognition by speech signals". In: *EuroSpeech*. 2003.

[Men+18] Z. Meng et al. "Speaker-invariant training via adversarial learning". In: *IEEE ICASSP*. 2018, pp. 5969–5973.

[Pan+15] V. Panayotov et al. "Librispeech: an ASR corpus based on public domain audio books". In: *IEEE ICASSP*. 2015, pp. 5206–5210.

[Pat12] M. A. Pathak. *Privacy-preserving machine learning for speech processing*. Springer Science & Business Media, 2012.

[Pov+11] D. Povey et al. *The Kaldi speech recognition toolkit*. Tech. rep. 2011.

[Pov+16] D. Povey et al. "Purely Sequence-Trained Neural Networks for ASR Based on Lattice-Free MMI". In: *INTERSPEECH*. 2016.

[PPK15] V. Peddinti, D. Povey, and S. Khudanpur. "A time delay neural network architecture for efficient modeling of long temporal contexts". In: *INTERSPEECH*. 2015, pp. 3214–3218.

[PSM14] J. Pennington, R. Socher, and C. Manning. "Glove: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543.

[Qia+17] J. Qian et al. "VoiceMask: Anonymize and Sanitize Voice Input on Mobile Devices". In: *CoRR* abs/1711.11460 (2017).

[Qia+18] J. Qian et al. "Hidebehind: Enjoy Voice Input with Voiceprint Unclonability and Anonymity". In: *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*. SenSys '18. Shenzhen, China: ACM, 2018, pp. 82–94.

[Rab+17] E. Rabinovich et al. "Personalized Machine Translation: Preserving Original Author Traits". In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp. 1074–1084.

[Rey95] D. A. Reynolds. "Speaker identification and verification using Gaussian mixture speaker models". In: *Speech Communication* 17.1-2 (1995), pp. 91–108.

[SB13] B. Schuller and A. Batliner. *Computational paralinguistics: emotion, affect and personality in speech and language processing*. John Wiley & Sons, 2013.

[Sch+06] J. Schler et al. "Effects of Age and Gender on Blogging." In: *Computational Approaches to Analyzing Weblogs, Papers from the 2006 AAAI Spring Symposium*. Stanford, CA, USA, Mar. 2006, pp. 199–205.

[Sch+13] B. Schuller et al. "The INTERSPEECH 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism". In: *Interspeech*. 2013, pp. 148–152.

[Sch+15] B. Schuller et al. "A survey on perceived speaker traits: Personality, likability, pathology, and the first challenge". In: *Computer Speech & Language* 29.1 (2015), pp. 100–131.

[Sek97] K. Sekiyama. "Cultural and linguistic factors in audiovisual speech processing: The McGurk effect in Chinese subjects". In: *Perception & Psychophysics* 59.1 (1997), pp. 73–80.

[Ser+16] D. Serdyuk et al. "Invariant representations for noisy speech recognition". In: *arXiv preprint arXiv:1612.01928* (2016).

[SN03] D. Sundermann and H. Ney. "VTLN-based voice conversion". In: *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (IEEE Cat. No.03EX795)*. 2003, pp. 556–559.

[Sny+18] D. Snyder et al. "X-vectors: Robust DNN embeddings for speaker recognition". In: *IEEE ICASSP*. 2018, pp. 5329–5333.

[Sri+19] B. M. L. Srivastava et al. "Privacy-Preserving Adversarial Representation Learning in ASR: Reality or Illusion?" In: *Annual Conference of the International Speech Communication Association (Interspeech)*. Graz, Austria, 2019.

[Sto+98] A. Stolcke et al. "Dialog act modeling for conversational speech". In: *AAAI Spring Symposium on Applying Machine Learning to Discourse Processing*. 1998, pp. 98–105.

[Tsu+18] T. Tsuchiya et al. "Speaker Invariant Feature Extraction for Zero-Resource Languages with Adversarial Learning". In: *IEEE ICASSP*. 2018, pp. 2381–2385.

[UK05] K. Umapathy and S. Krishnan. "Feature analysis of pathological speech signals using local discriminant bases technique". In: *Medical and Biological Engineering and Computing* 43.4 (2005), pp. 457–464.

[VK04] D. Ververidis and C. Kotropoulos. "Automatic speech classification to five emotional states based on gender information". In: *EUSIPCO*. 2004, pp. 341–344.

[VPB09] A. Vinciarelli, M. Pantic, and H. Bourlard. "Social signal processing: Survey of an emerging domain". In: *Image and Vision Computing* 27.12 (2009), pp. 1743–1759.

[Wah00] W. Wahlster, ed. *Verbmobil: Foundations of Speech-to-Speech Translation*. Berlin: Springer, 2000.

[Wat+17] S. Watanabe et al. "Hybrid CTC/attention architecture for end-to-end speech recognition". In: *IEEE Journal of Selected Topics in Signal Processing* 11.8 (2017), pp. 1240–1253.

[Wat+18] S. Watanabe et al. “ESPnet: End-to-End Speech Processing Toolkit”. In: *Interspeech*. 2018, pp. 2207–2211.

[WRS02] K. Weilhammer, U. Reichel, and F. Schiel. “Multi-Tier Annotations in the Verbmobil Corpus”. In: *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002)*. Las Palmas, Canary Islands, Spain, May 2002.

[Zen+06] Y.-M. Zeng et al. “Robust GMM based gender classification using pitch and RASTA-PLP parameters of speech”. In: *International Conference on Machine Learning and Cybernetics*. 2006, pp. 3376–3379.

A Appendix

A.1 Personal data categorisation

Categories	**Words/Phrases**
Knowledge and Belief	Communist, fascist, liberal, leftist Christian, muslim,jewish,,atheist, Aesthetic,nihilist , pagan I think that _____ I know that _____ I believe in _____ My philosophy is _____ My religion is _____
Authenticating	Card number _____ Access word _____ Secret word _____ Password _____ PIN
Preference	I like _____ In my opinion _____ I prefer _____ I'm interested in _____ I allways buy _____ I want. _____ My favourite _____
Life History	When my (relative/ friend...) died _____ _____ happened to me When I was in the university _____ When I was a kid _____ During my last holidays _____
Account	PIN _____ Card Holder _____ Short code _____ CVV _____ Credit card
Ownership	Car brand _____ my house is in _____ my garage is in _____ I have rented _____ I borrowed _____ I have a _____ I own _____
Transactional	I've received _____ euro/dollars/..... I've paid _____ euros/ dollars/..... I 've charged _____ euros/dollars/.... I bought it for _____ euros/dollars/....

	I was paid _____ euros/dollars/...
Credit	Balance Loan Red numbers Payroll Mortgage I have _____ euros Broke
Identifying	Identification document, passport, driving license My photo Name, surrname, family name Nickname My name is _____ I'm known as _____ They call me _____
Ethnicity	Indo-european Turkic Caucasic Austronesian Basque English German Spanish
Sexual	Heterosexual, straight ,homosexual, gay, lesbian Vondage, sado, porn, fetiche Travestite, transgendered Asexual Hermaphrodite
Behavioural	I visit _____ website I buy through _____ I practice _____ I usually _____ I allways _____
Demographic	My age is _____ Single, married, divorced Middle class, high class, working class Young, mature, old Employed, unemployed, retired
Medical and Health	Insurance Disease, analysis. Deaf, blind, phisical disability, mental disability. Bipolar, depresion, schizophrenia Pills, tablets, anthibiotic, treatment The test results show _____ I've been prescribed _____

	I suffer from _____
Physical Characteristic	blonde, brown, redhead I'm ... years old Man, woman, male, female Tall, short skinny, fat
Professional	Lawyer, police, doctor, commercial Directive, manager, senior , trainee Certificate, Master, speciality I was dismissed from _____ I work in _____
Criminal	Robery, burglary, heist. Parole, sentence, conviction, sentence, . Lawyers, assistance, court. Appeal, pardons. Rights.
Public Life	Married, single Atheist, Catholic, Jewish, Muslim Good, Nice, disagreeable, nasty, unfriendly republican, conservative I have a group of .....
Family	Wife, husband, son, daughters, girlfirend Married, divorce, My wife _____ My boyfriend _____ I'm married with _____ I'm going out with _____
Social Network	My friends are _____ My contacts are _____ I'm member of _____ Association LinkedIn , Facebook , Twitter, Instagram, Flirck, Whatsapp, YouTube, Tumblr...
Communication	Group members are _____ My phone book My email is _____ I'm in Facebook/ Twitter/ LinkedIn My whatsapp _____ Messages
Computer Device	IP adress Device ID
Contact	My e mail adress is _____. Telephone mumber. _____ Social network _____ Pager number. _____

	Address (home and work) _____
Location	Club, restaurant, hospital, shop, cafe altitude and latitude My address is _____ I live in _____ I am in _____ I'm close to _____ I'm visiting _____

A.2 Categorisation list of personal data: sources

A.2.1 Recitals (14), (15),(26) and (30) and Articles 2, 4(1) and 9 of the GDPR

In this first section we report the recitals and excerpts of the articles of the GDPR from which we can define the concept of personal data.

> (14) The protection afforded by this Regulation should apply to natural persons, whatever their nationality or place of residence, in relation to the processing of their personal data. This Regulation does not cover the processing of personal data which concerns legal persons and in particular undertakings established as legal persons, including the name and the form of the legal person and the contact details of the legal person.
>
> (15) In order to prevent creating a serious risk of circumvention, the protection of natural persons should be technologically neutral and should not depend on the techniques used. The protection of natural persons should apply to the processing of personal data by automated means, as well as to manual processing, if the personal data are contained or are intended to be contained in a filing system. Files or sets of files, as well as their cover pages, which are not structured according to specific criteria should not fall within the scope of this Regulation.
>
> (26) The principles of data protection should apply to any information concerning an identified or identifiable natural person. Personal data which have undergone pseudonymisation, which could be attributed to a natural person by the use of additional information should be considered to be information on an identifiable natural person. To determine whether a natural person is identifiable, account should be taken of all the means reasonably likely to be used, such as singling out, either by the controller or by another person to identify the natural person directly or indirectly. To ascertain whether means are reasonably likely to be used to identify the natural person, account should be taken of all objective factors, such as the costs of and the amount of time required for identification, taking into consideration the available technology at the time of the processing and technological developments. The principles of data protection should therefore not apply to anonymous information, namely information which does not relate to an identified or identifiable natural person or to personal data rendered anonymous in such a manner that the data subject is not or no longer identifiable. This Regulation does not therefore concern the processing of such anonymous information, including for statistical or research purposes.
>
> (30) Natural persons may be associated with online identifiers provided by their devices, applications, tools and protocols, such as internet protocol addresses, cookie identifiers

> or other identifiers such as radio frequency identification tags. This may leave traces which, in particular when combined with unique identifiers and other information received by the servers, may be used to create profiles of the natural persons and identify them.
>
> Article 2: Material scope 1.This Regulation applies to the processing of personal data wholly or partly by automated means and to the processing other than by automated means of personal data which form part of a filing system or are intended to form part of a filing system. [...]
>
> Article 4: Definitions: [...] (1) "personal data" means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person; [...]

Article 9: Definitions: Processing of special categories of personal data.

> 1. Processing of personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership, and the processing of genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person's sex life or sexual orientation shall be prohibited. [...]

As can be seen, the GDPR includes some possible categories as examples in Art. 4.1: "name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person" In Art. 9 the GDPR provides a concrete categorization of the "special categories of personal data" for sensitive data. All these categories are included in the categorization model chosen for the list.

A.2.2 Art.29 Data Protection Working Party: WP 01245/07/EN, WP 136 Opinion 4/2007 on the concept of personal data

We report below some excerpts of the Opinion 4/2007 issued by the Art. 29 Working Party (replaced on 25 of May of 2018 by the European Data Protection Board) that help understand the concept of personal data and its possible categorizations.

> [...] It needs to be noted that this definition reflects the intention of the European lawmaker for a wide notion of "personal data" [...]
>
> [...] The term "any information" contained in the Directive clearly signals the willingness of the legislator to design a broad concept of personal data. This wording calls for a wide interpretation. [...]
>
> [...] From the point of view of the nature of the information, the concept of personal data includes any sort of statements about a person. It covers "objective" information, such as the presence of a certain substance in one's blood. It also includes "subjective" information, opinions or assessments. [...]

[...] From the point of view of the content of the information, the concept of personal data includes data providing any sort of information. [...]

[...] The term "personal data" includes information touching the individual's private and family life "stricto sensu", but also information regarding whatever types of activity is undertaken by the individual, like that concerning working relations or the economic or social behaviour of the individual. It includes therefore information on individuals, regardless of the position or capacity of those persons (as consumer, patient, employee, customer, etc). [...]

[...] Considering the format or the medium on which that information is contained, the concept of personal data includes information available in whatever form, be it alphabetical, numerical, graphical, photographical or acoustic, [...]

[...] Special reference should be made here to biometric data These data may be defined as biological properties, physiological characteristics, living traits or repeatable actions where those features and/or actions are both unique to that individual and measurable, even if the patterns used in practice to technically measure them involve a certain degree of probability. Typical examples of such biometric data are provided by fingerprints, retinal patterns, facial structure, voices, but also hand geometry, vein patterns or even some deeply ingrained skill or other behavioural characteristic [...]

[...] In general terms, information can be considered to "relate" to an individual when it is about that individual[...]

[...] In many situations, this relationship can be easily established[...]

[...] A number of other situations can be mentioned, though, where it is not always as self evident as in the previous cases to determine that the information "relates" to an individual. [...]

[...] In general terms, a natural person can be considered as "identified" when, within a group of persons, he or she is "distinguished" from all other members of the group[...]

[...] Accordingly, the natural person is "identifiable" when, although the person has not been identified yet, it is possible to do it[...]

[...] Identification is normally achieved through particular pieces of information which we may call "identifiers" and which hold a particularly privileged and close relationship with the particular individua. Examples are outward signs of the appearance of this person, like height, hair colour, clothing, etc. or a quality of the person which cannot be immediately perceived, like a profession, a function, a name etc. [...]

[...] Further clarification is contained in the commentary to the Articles of the amended Commission proposal, in the sense that "a person may be identified directly by name or -13- indirectly by a telephone number, a car registration number, a social security number, a passport number or by a combination of significant criteria which allows him to be recognized by narrowing down the group to which he belongs (age, occupation, place of residence, etc.)". The terms of this statement clearly indicate that the extent to which certain identifiers are sufficient to achieve identification is something dependent on the context of the particular situation. [...]

A.3 Kaldi ASR recipe for Verbmobil

This is a Kaldi recipe for training speech recognition models on the English subset of the Verbmobil corpus. The objective to is to train Kaldi's nnet3 and chain models.

Table of Contents

Prerequisites

- This recipe will re-use binaries and scripts from the Kaldi toolkit. So you should have Kaldi pre-installed on your system.
- It requires you to install the kaldi_lm tool. You can install this tool with the *tools/extras/install_kaldi_lm.sh* script in your Kaldi installation.
- A dump of the BAS edition of the English dialogs in Verbmobil I corpus and that of the English dialogs in Verbmobil II. This implies CDs 6, 8 and 13 for Verbmobil I, and CDs 23, 28, 31, 32, 42, 43, 47, 50, 51, 52, 55, 56 for Verbmobil II.

Setup

- Ensure that you have a working Kaldi installation.
- Modify the softlinks *steps/* and *utils/* in this directory to point to *egs/wsj/s5/steps/* and *egs/wsj/s5/utils/*, respectively, in your Kaldi installation.
- Modify the path of *KALDI_ROOT* and modify (or remove) the path to *kaldi_lm* and *sox* tools in path.sh
- Modify cmd.sh if you are using a different execution queue for Kaldi.

Running the recipe

The Kaldi Verbmobil English recipe has currently three main stages: 1. Data Preparation: To process the original BAS edition of the corpus and prepare

data for following stages. 2. Train GMM HMM models: To train the initial GMM-HMM models on the corpus. 3. Train nnet3 models: To train nnet3 chain models using the initial GMM-HMM models.

Data Preparation

This stage processes the original BAS edition of the corpus to prepare data for following training stages. The script for data preparation is run as:

```
bash prepare_data.sh <trl_dir>
```

`<trl_dir> is the parent directory containing all *.trl* transcriptions (for all dialogs in all Verbmobil CDs)`

After running the data preparation script above please ensure that:

- *data/local/dict/lexicon.txt.orig* lexicon file does not have missing pronunciations. Search for '' tags and add the missing pronunciations manually. You can also use the online CMU Lexicon Tool to generate these pronunciations. This step would be ideally replaced by an automatic G2P tool in the future.
- *data/local/dict/lexicon.txt.orig* file, whether modified or not, is moved/copied to *data/local/dict/lexicon.txt*.
- contents of *data/local/dict/*, including files *optional_silence.txt extra_questions.txt nonsilence_phones.txt silence_phones.txt*, are verified.

Train GMM HMM models

This stage trains the initial GMM-HMM (tri-phone) models on the Verbmobil corpus. The training script is launched as:

```
bash train_tri3.sh
```

Note that the script train_tri3.sh has some pre-defined flags and constants including:

- flags to manage different stages in training tri-phone models
- constants defining number of jobs to be used in feature extraction, training and decode
- constants for tri-phone model parameters
- constants for the subset of data used for training the initial mono-phone and tri-phone models
- paths and prefixes to the directory containing Verbmobil signal files

Train nnet3 models

After training the initial GMM-HMM (tri-phone) models on the Verbmobil corpus you can go training an nnet3 TDNN model or an nnet3 TDNN Chain model.

An nnet3 TDNN model can be trained using:

```
bash local/nnet3/run_tdnn_1a.sh
```

Instead of the above nnet3 TDNN model, one can train an nnet3 TDNN Chain model using:

```
bash local/chain/run_tdnn.sh
```

Evaluation scripts

- *test_tr3.sh* to decode the test set with tri3 GMM HMM model
- *local/nnet3/test_nnet3.sh* to decode the test set with nnet3 model
- *local/chain/test_chain.sh* to decode the test set with nnet3 chain model
- *local/gen_ctm.sh* to generate CTM output from a decode directory
- *local/ctm2txt.pl* to generate ASR transcripts from CTM output
- *local/analyse-wer.sh* and *analyse-wer-VNC-ADB.sh* to calcuate WER performance of a particular model, as shown in WER table below

Note: Evaluation scripts are to be run individually and may require creating the expected output directories.

Additional Notes

Corpus and Data

- Currently the recipe has been tested on the English subset of Verbmobil corpus in the dump at */talc/multispeech/corpus/dialog/Verbmobil/Verbmobil* on Nancy site of Grid5000.
- sample *data/* and *exp/* directories and associated log files are available at */talc3/multispeech/calcul/users/isheikh/exp/vm-recipe/* on Nancy site of Grid5000.

train-dev-test split

The current train-dev-test split has 92 dialogs in test, 21 dialogs in dev and 698 dialogs in train, ensuring that a new dialog domain and German English speakers are seen in dev and test, without any dev/test speakers seen in train. This splitting is implemented in the local/getTrainDevTestSplits.pl script. Based on the corpus stats, it uses the following approach to decide the split:

- All dialogs in *CDs 47, 52, 55, 56* (including *info desk* domain + German English speakers) and dialogs *q010-q020* from *CD 6* (including German English speakers) are put into the test set. Additonally, dialogs from other CDs including the speakers from test set are also put into the test set.
- Dialogs in *CDs 32, 51* (including *info desk* domain + German English speakers) and dialogs *q001-q009 q021 q0022* from *CD 6* (including German English speakers), which are not part of the test set, are put into the dev set. Additonally, dialogs from other CDs including the speakers from dev set are also put into the dev set. Moreover, to ensure a balance in the WERs of test and dev dailogs corresponding to speakers *HCF QZO QZB* are moved from test to dev.
- All remaining dialogs are kept as the train set. Except for dialogs of speakers *ADB* (British English speaker) and *VNC* (with Asian accented English), which are put into test and dev, respectively.

Problem with speaker ids

As highlighted in the corpus stats, speaker ids used in the Verbmobil coprus are not unique. After manually listening to dialogs, it was identified that

- speakers ids *CAC JDH NKH RGM* represent different speakers in Verbmobil parts I and II
- speaker id *MAS* represents two different speakers within Verbmobil part I To address this issue,
- speakers ids *CAC JDH NKH RGM* in Verbmobil parts I and II are suffixed with *VM1* and *VM2*, respectively
- speaker id *MAS* for dialog *q007n* in Verbmobil part I is suffixed with *VMZ*
- all other speaker ids are suffixed with *VMX*

WER with tri3 and nnet3 models

WER obtained for train-dev-test split discussed above:

	EN+DE spkrs	EN spkrs	DE spkrs	Domain A	Domain B	A n EN
dev # wrd	18480	12706	5774	12861	5619	7087
dev # sent	1113	784	329	794	319	465
tri3 dev si	35.84	31.50	45.38	34.33	39.30	25.33
tri3 dev	31.68	26.87	42.28	30.57	34.24	21.02
nnet3 dev	26.16	21.83	35.69	25.58	27.50	17.34
chain dev	21.87	17.39	31.73	22.34	20.79	14.69
test #wrd	36222	28885	7337	23199	13023	15862
test #sent	2286	1971	315	1383	903	1068
tri3 test si	34.71	31.54	47.17	35.26	33.72	29.75
tri3 test	28.86	25.63	41.57	29.86	27.08	24.44
nnet3 test	25.45	22.93	35.37	25.71	24.99	21.24
chain test	22.36	20.03	31.54	22.45	22.19	18.25

where 'EN' and 'DE' denote American and German English speakers, 'DA' and 'DB' denote *appointment scheduling* and *info desk* dialog domains, respectively, as per the corpus annotation. 'si' denotes speaker independent decoding.

WER obtained for train-dev-test split discussed above and also separting out the accented speakers, *ADB* (British English speaker) and *VNC* (with Asian accented English), into the 'DE' group:

	EN+De spkrs	EN- spkrs	DE+ spkrs.	DA n EN-
dev # wrd	18480	12049	6431	6430
dev # sent	1113	706	407	387
tri3 dev si	35.84	32.04	42.95	25.71
tri3 dev	31.68	27.26	39.96	21.17
nnet3 dev	26.16	22.09	33.79	17.37
chain dev	21.87	17.45	30.14	14.54
test #wrd	36222	26621	9601	13598
test #sent	2286	1817	469	914
tri3 test si	34.71	30.51	46.36	27.42
tri3 test	28.86	24.60	40.66	22.23
nnet3 test	25.45	22.28	34.25	19.68
chain test	22.36	19.45	30.42	16.83

A.4 Speech and text alignment tool

This code does forced alignment of a set of speech files and their corresponding text transcriptions. It generates two type of alignments: (1) alignment at the word level, and (b) alignment at the phone level.

Please go through prerequisites and assumptions before going on to the setup and usage of this code .

Prerequisites and Assumptions

- The code re-uses binaries and scripts from the Kaldi toolkit. So you should have Kaldi pre-installed on your system.
- The code requires that you have a pre-trained Kaldi GMM acoustic model and a corresponding `lang/` directory. The Librispeech speech model shared along with this code can be used for your purpose, provided that you are dealing with clean 16kHz speech data. Otherwise you can choose a suitable pre-trained Kaldi GMM model from the Kaldi branches

Setup

- Ensure that you have a working Kaldi installation.
- Modify the softlinks `steps/` and `utils/` in this directory to point to `egs/wsj/s5/steps/` and `egs/wsj/s5/utils`, respectively, in your Kaldi installation.
- Modify the path of KALDI_ROOT and modify/remove the path to flac utility in path.sh
- Modify cmd.sh if you are using a different execution queue for Kaldi.
- Modify files in conf/ as per your pre-trained Kaldi GMM acoustic model.

Usage

- Ensure that you have completed the steps in the setup section above.
- Similar to Kaldi acoustic model training, prepare a data/ directory containing the following files:

```
'text' file containing audio file id and its corresponding \
    cleaned text transcription, one per line.\
'wav.scp' file listing the audio file ids and the corresponding \
    audio file access descriptor.\
'utt2spk' file mapping utterance ids to speaker ids\
'spk2utt' file mapping speaker ids to utterance ids
```

  For examples of these files refer Kaldi data preparation or look into `task_libri/data/test_clean/` directory shared separately with this code.

- Verify the number of jobs specified in the beginning of align.sh script
- Launch the alignment script as follows:


```
bash align.sh <data_dir> <mdl_dir> <lang_dir> <ali_dir>"

data_dir directory containing text wav.scp utt2spk spk2utt files\
mdl_dir directory with the acoustic model for alignment\
lang_dir directory for alignment\
ali_dir is output directory for output alignments
```

For example, to try out using the Librispeech model and data shared along with this code:

```
bash align.sh task_libri/data/test_clean/ task_libri/model/tri6b_vassil/
task_libri/model/lang_vassil/ task_libri/ali/test_clean/
2>&1 | tee task_libri/align_test-clean.log
```

After successful completion, this script should create phone level and word level alignments in CTM format in `<ali_dir>/phn_ali/` and `<ali_dir>/wrd_ali/` directories, respectively.

Note

- The current version does not explicitly handle non-verbal lexical entities and non-silence phones like [noise], [laughter], etc.
- The number of jobs used by the alignment process is hardcoded to 16 in align.sh.
- `task_libri` sample data is available at `/talc3/multispeech/calcul/users/isheikh/exp/align-s2t/task_libri` on Nancy site of Grid5000.

A.5 Text Transformer tool

The text transformer tool helps to:

- identify sensitive words or named entities in a dialog conversation
- Text transformation of sensitive words by words of the same-type or placeholders

---

Requirements

Python 2.7 or 3 with following packages:

- numpy
- pandas
- flair

Usage

Identification of named entities in a dialog conversation

Given an annotated text file(s) in a CoNLL format, split them into training, development and text split using:

python preprocess.py

The named entity recognition (NER) is trained using Flair that is based on BiLSTM-CRF model and word features are obtained from Glove embeddings.

To train the NER model, use:

python train_ner.py —input_dir training_data_dir —output_dir model_output_dir

`training_data_dir` the directory consisting of the training, development and test data in '.tsv' extension

`model_output_dir` the directory where the model output, and training log is stored

To test the NER model, use:

python test_ner.py

Text transformation of sensitive words by words of the same-type or placeholders

1) create training dataset to evaluate the impact of different text transformation strategies, we consider 5 ways of transforming sensitive words/ named entities (PER, LOC, ORG, DATE and TIME):

- replace all single words labeled as named entities with a placeholder
- replace all multi-word expressions labeled as named entities with a placeholder

- replace all single words labeled as named entities with the same-type word
- replace all multi-word expressions labeled as named entities with same-type word
- replace all multi-word expressions labeled as named entities with a multi-word expression of the same-type

python create_privacy_transformed_data.py

2) Demo of text transformation, that accepts a sentence from the command line, automatically identify the sensitive words, and replace each single word labeled as named entity with another word of the same-type

python demo_text_transformation.py

Enter a sample sentence: Mark Smith is going to London in April

Tagged sentence: Mark Smith is going to London in April

Transformed sentence: Bottermaker Norbert is going to London in Thursday

A.6 Voice Transformer tool

This project provides:

- classes and functions to perform voice conversion in the voice_transformation library
- and a script to apply them on Librispeech or Verbmobil corpus.

In this version, 2 voice conversion techniques are available in dedicated modules: voicemask (inspired by [1] and [2]) and a vtln-based method (inspired by [3])

For each of these methods, the module consists of:

- a `builder` function to compute some params, specific to each method, from target speakers data.
- a `Transformer` class to transform the utterances of a speaker. This class has to be:
  - initialized with the pre-built params (and eventual additional parameters)
  - fitted with the speaker voice

In the COMPRISE use case:

- the builder would be use to pre-build the parameters to be embedded in the app
- the Transformer class is the part that would be ran on the device

[1] Qian, J., Du, H., Hou, J., Chen, L., Jung, T., Li, X. Y., ... & Deng, Y. (2017). Voicemask: Anonymize and sanitize voice input on mobile devices. arXiv preprint arXiv:1711.11460.
[2] 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). ACM.
[3] 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.

Install

Requirements:

- numpy
- scipy
- soundfile
- pyworld

Using a conda environment :

```
conda create --name speech scipy numpy tqdm scikit-learn
conda activate speech
pip install soundfile pyworld
```


Script

A script is available at the root of this repository to run the transformations with pre-defined settings on
Librispeech or Verbmobil.
With this script, each utterance of each speaker is converted to an arbitrary target speaker among a group of randomly chosen speakers

Quickstart with voicemask on librispeech

```
python apply_transformation.py voicemask\
       librispeech $LIBRISPEECH_ROOT/dev-clean
```

This will :

- create a `output` dir in the current directory
- create a subdirectory `dev-clean__mod_voicemask` in it
- replicate the structure (chapter and speakers) of the original `$LIBRISPEECH_ROOT/dev-clean` in the new `output/dev-clean__mod_voicemask` folder
- process all the utterances of the dev-clean subset of the Librispeech corpus, and save them in this structure

Positional arguments

The first positional argument is the method to use : `voicemask` or `vtln`.
The second one is the name of the corpus (`librispeech` or `verbmobil`).
The following positional arguments are the paths to the subsets to transform (you can specify several of them)

Optional arguments

- `-T NB_TARGETS` : the maximum number of target speakers to choose. Since the data of the target speakers are kept in memory, this number must be chosen to fit in memory

- `-N nb_proc` option lets you decide how many jobs you want to use for the parallelized portions of the code.
- `-o output_path` : where to save the transformed utterances. Default is an `output` folder created in the current directory.

- `-s suffix` : which suffix to add to the original subset name. Default is `_mod_voicemask` for voicemask, `_mod_vtln` for the VTLN-based conversion
- `--resume` flag let you resume a previously interrupted run
- `--targets_file TARGETS_FILE` : path to a previously created target file to use the same mapping (useful with `--resume`)

A.7 Word Masking tool

The word masking tool helps to:

- mask sensitive words in speech files
- mask sensitive words in speech transcription files

Requirements

Python 2.7 or 3 with following packages:

- numpy
- scipy
- tqdm

Usage

Masking sensitive words in a speech file

Given a set of speech files, their corresponding word level transcriptions and the word level tags the following script helps to mask all the sensitive words in all the speech files:

```
python mask_words_in_speech.py word_tag_file \ tag_format
tag_csv corpus corpus_path \ time_alignments output_path
```

- `word_tag_file` contains a list of all words in all the expected utterances. (Currently supoorts only the CoNLL NER format with only NER tags.)
- `tag_format` is the format of the word_tag_file file. (Currently supports only **conll_ner**)
- `tag_csv` is the csv of tags to be considered as sensitive words e.g. PER,LOC,ORG
- `corpus` is the name of the speech corpus. (Currently only 'verbmobil' is supported.)
- `corpus_path` is the path to the speech corpus
- `time_alignments` is the directory containing the time alignments. (Currently only CTM format is supported.)
- `output_path` is the output directory where the masked speech files will be created

The script will create a **wav** directory containing the masked speech files and a Kaldi style *wav.scp* file.

Masking sensitive words in a speech file

Given a file with word level tags the following script helps to mask all the sensitive words in all the speech transcription files:

```
mask_words_in_transcripts.py word_tag_file tag_format\
tag_csv corpus output_file
```


- `word_tag_file` contains a list of all words in all the expected utterances. (Currently supoorts only the CoNLL NER format with only NER tags.)
- `tag_format` is the format of the word_tag_file file. (Currently supports only **conll_ner**)
- `tag_csv` is the csv of tags to be considered as sensitive words e.g. PER,LOC,ORG
- `corpus` is the name of the speech corpus. (Currently only 'verbmobil' is supported.)
- `output_file` is output word sensitive words masked transcription file

The output file is a Kaldi style *text* transcription file.