Prεεch: A System for Privacy-Preserving Speech Transcription

Shimaa Ahmed, Amrita Roy Chowdhury, and Kassem Fawaz, and Parmesh Ramanathan,
University of Wisconsin—Madison

This paper is included in the Proceedings of the
29th USENIX Security Symposium.
August 12–14, 2020

Contents

I. Introduction

II. Prεεch Overview

III. Prεεch

IV. Evaluation

V. Conclusion

1. Introduction

- 음성데이터에는 여러 민감 정보가 있음.

  ex) 민감정보: 성별, 화자의 감정 등. 텍스트 중 민감정보(개인정보): 이름, 위치(주소), 건강정보 등
- 오프라인 ASR(Automatic Speech Recognition)은 민감정보가 유출될 위험성↓, 성능 ↓
- 온라인 ASR(google 혹은 아마존 제공 ASR API)은 민감정보가 유출될 위험성↑, 성능 ↑
- **오프라인 ASR처럼 민감정보가 유출될 위험성은 줄이고, 온라인 ASR 만큼의 좋은 성능을 가지는 ASR 시스템을 만들고자 함.**

1. Introduction

Prεεch provides an end-to-end tunable system

Prεεch의 목표

1) **protect the users' privacy along the acoustic and textual dimensions**
   - 음향학적 정보: 성별, 감정 등
   - 텍스트 정보: 의료정보, 이름, 위치 등
2) **improve on the transcription accuracy compared to offline models**
   - 기존 오프라인 모델 (ex. deepspeech)보다 높은 transcription accuracy
3) **provide the users with control knobs to customize Prεεch's functionality according to their desired level of utility, usability, and privacy.**
   - control knobs 를 통해 prεεch의 기능을 사용자정의할 수 있게 하여 사용자가 원하는 레벨의 효용성, 사용성, 개인정보보호 제공

2. Prεεch Overview

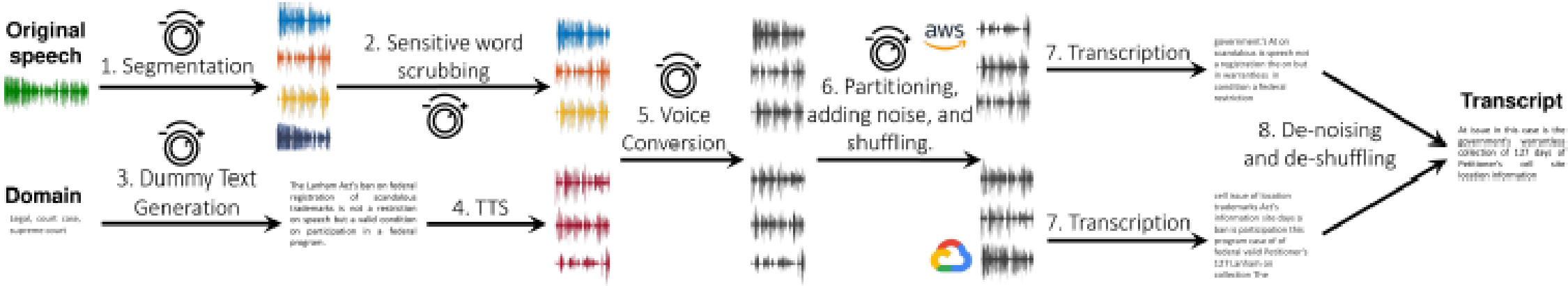


Figure 1: High-level overview of Prεεch, showing the knobs where a user can tune the associated trade-offs.

1. 원본 스피치를 한 단어로 분할
2. 민감 단어 삭제
3. 도메인을 정함. (어떤 정보인지)
   -> 도메인과 관련된 더미 텍스트를 생성함.
4. 생성된 더미 text 를 speech 형태로 만듦.
5. 음성변환을 적용시킴.
6. 분할, 노이즈 더함, 더미와 원본 섞음.
7. transcription 생성
8. 노이즈 없애고 셔플 없애서 다듬어서 user에게 전달되는 transcript 를 만듦.

3. Prεεch

1. Segmentation

- Speech S 의 context 를 없애기 위해 segment $\mathbb{S}$ 로 분할
- 계층적 segmentation 기법 적용

  1) silence detection 2) pitch detection
- 각 segment 는 여러 단어를 포함할 수 있음.
- pitch 정보를 감지하여 segment 를 더욱 미세하게 분할
  - non-speech 감지
  - 각 segment는 20ms 이상, 가까이 있는 세그먼트 통합
- Control Knobs:
  - segment $\mathbb{S}$ 의 최소길이 조정 가능
  - segment $\mathbb{S}$ 가 작을수록 –〉 context 손실, transcript 정확도 저하, Privacy 향상

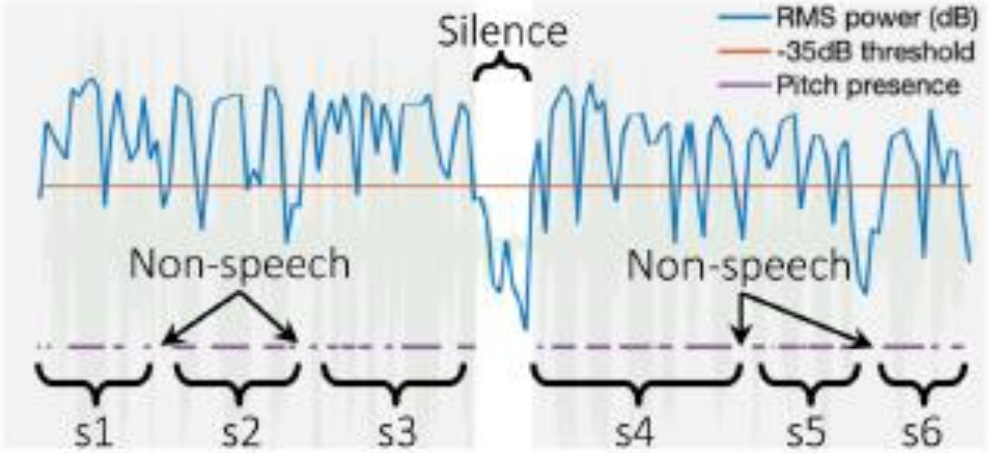


Figure 2: An illustration of Prεεch's segmentation algorithm. The coarse segments in light gray. The absence of pitch information indicate non-speech instances, which further breaks down the coarse segments into finer segments.

3. Prεεch

2. Sensitive Words Scrubbing

Offline Service Provider

- Speech S의 offline transcript T_S^{OSP} 를 얻음.
- T_S^{OSP} 에 NER(Named Entity Recognition) 적용하여 text에서 named entity 를 찾아 사람, 조직, 위치 등의 범주로 나눔
- 사용자 정의 민감 단어 목록(sensitive words) 을 만든 후, segment $\mathbb{S}$ 에 KWS(KeyWord Spotting) 을 적용.
  - KWS(KeyWord Spotting) : NER로 식별할 수 없는 사용자 정의 키워드를 찾는데 사용됨.
- Control Knobs: KWS 는 사용자 정의 키워드 목록을 가져와 민감도 점수에 따라 음성 파일에서 키워드 판별
  - 민감도 점수: 키워드 발견에 필요한 음성 유사성에 대한 threshold
  - 민감도 점수 ↓: 해당 단어와 비슷한 단어에 모두 flag 지정
  - 민감도 점수 ↑: 해당 단어와 완벽히 일치한 단어에 flag 지정 (누락율 ↑)
  - 따라서 민감도 점수는 privacy 와 utility 간의 trade-off hyperparameter 이다.

3. Prεεch

3. Dummy segment 생성 mechanism – Differential Private Word Histogram

- vocabulary $\mathcal{V}$ 정의
  Ground truth transcription
  : T_S^g 를 구성하는 non-stop and stemmed words (stop words 가 아닌 단어들과 어원 단어들)
- T_S^g 에서 $w_i \in \mathcal{V}$ 의 등장 빈도수 c_i
- BoW(Bag of Words): $\{w_i : c_i | w_i \in \mathcal{V}\}$ ex. {'ab': 3, 'abs': 1, ... }
- H : $[c_i]$ BoW의 count vector

3. Prεεch

3. Dummy segment 생성 mechanism – Differential Private Word Histogram

Privacy Definition

- 어떤 단어가 많이 나오는지에 대한 정보는 민감정보이기 때문에 프라이버시 보호 방식으로만 CSP(Cloud Service Provider)에 공개할 수 있음.
- 프라이버시 보호 방식으로 DP(Differential Privacy) 채택

Definition 4.1 ((ε,δ)-differentially private d-distant histogram release). A randomized mechanism $\mathcal{A}: \mathbb{N}^{|\mathcal{V}|} \rightarrow \mathbb{N}^{|\mathcal{V}|}$, which maps the original histogram into a noisy one, satisfies (ε,δ)-DP if for any pair of histograms H_1 and H_2 such that $||H_1 - H_2||_1 = d$ and any set $O \subseteq \mathbb{N}^{|\mathcal{V}|}$,

$$Pr[\mathcal{A}(H_1) \in O] \leq e^{\varepsilon} \cdot Pr[\mathcal{A}(H_2) \in O] + \delta. \qquad (1)$$

어떤 $\mathbb{N}^{|\mathcal{V}|}$에 속하는 임의의 집합 O와 히스토그램 H1과 H2의 L1 distance 가 d인 페어에 대해 (e,d)-DP를 만족하도록 랜덤하게 Nv->Nv 로 매핑하는 메커니즘 A

- CSP 관점에서는 noisy histogram $\widetilde{H}$(H와 d만큼 차이남) 를 보게 됨.

3. Prεεch

3. Dummy segment 생성 mechanism – Differential Private Word Histogram

Privacy Definition

- 많이 나오는 단어로 정보 유추 가능

(a) Original (b) ε=1, δ=0.05, and d=2 (c) ε=1, δ=0.05, and d=5 (d) ε=1, δ=0.05, and d=10

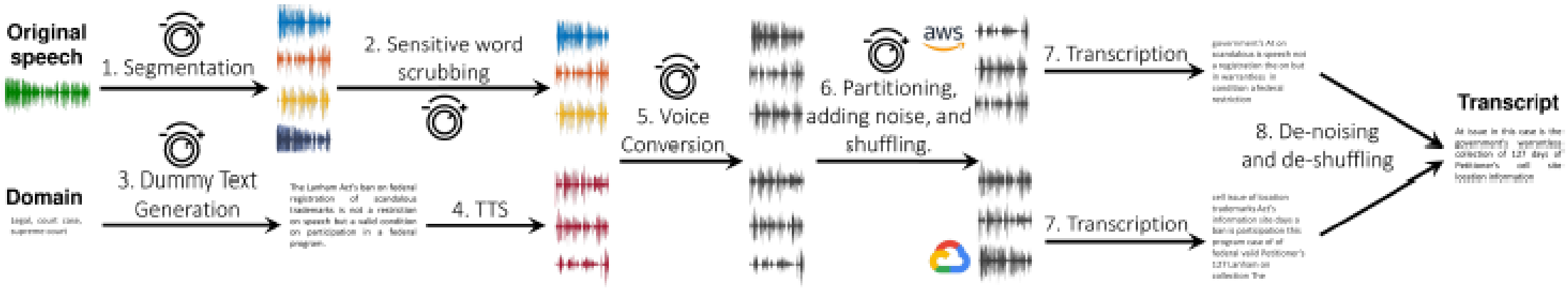
Figure 3: The word cloud of the Facebook dataset visualizing the histogram as it changes after adding different levels of noise.

3. Prεεch

3. Dummy segment 생성 mechanism

DP mechanism by which Prεεch generates the dummy segments for S.

- mechanism 에 대한 입력
  - (1) $\mathbb{S}$: 원본 speech S에 대한 짧은 segment
  - (2) The privacy parameter ε and δ
  - (3) N – the number of non-colluding CSPs to use.



3. Prεεch

3. Dummy segment 생성 mechanism

DP mechanism by which Preech generates the dummy segments for S.

- mechanism 동작방식
  - S에 대한 offline transcript 를 생성하는 동안 vocabulary $\mathcal{V}$ 를 정의

    (T_S^{OSP}에서 m 퍼센트 이상 등장한 단어 ∪ TF-IDF 값이 Δ(in T_S^{OSP}) 보다 큰 단어)
  - d의 값 조정 (특정 단어와 얼마나 비슷한 단어인지 Original과 noisy 의 거리 d. Thm 4.4)
  - N개의 개별 noise vector 생성, $\eta_i \sim [Lp((\ln(1+\frac{1}{\beta}(e^{\varepsilon}-1)), \beta\delta, d)]^{|\mathcal{V}|}, i \in [N].$

    개별 노이즈 벡터는 라플라스 분포를 따르도록 생성.

    partition i 에 대해 $\mathcal{V}$ 의 각 단어를 음이 아닌 정수 noise 값과 연관 시킴.
  - NLP에서 생성된 텍스트에서 $\mathcal{V}$ 의 단어를 포함하는 모든 text segment 를 추출

    사용하는 corpus 의 text segment 를 샘플링하여 noise vector η_i와 match

    partition i 에 대한 noise (dummy) segments : $\mathbb{S}_{d,i}$

    필요한 noise count 가 충족될 때까지 NLP 언어 모델에서 텍스트를 생성하는 과정을 반복

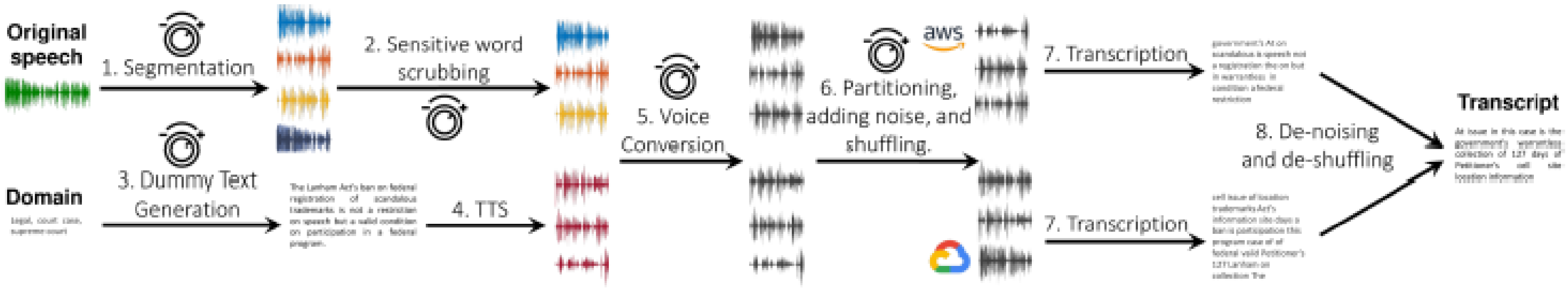
3. Prεεch

3. Dummy segment 생성 mechanism

DP mechanism by which Preech generates the dummy segments for S.

- mechanism 동작방식
  - $\mathbb{S}$를 랜덤하게 N개의 집합으로 분할합니다. $\mathbb{S}_i$, $i \in [N]$

    $\Pr[\text{segment } s \text{ goes to partition } i] = \beta = 1/N, s \in \mathbb{S}.$
  - 각 partition $i \in [N]$ 에 대해, TTS와 VC 이후에 $\mathbb{S}_{d,i}$ 와 $\mathbb{S}_i$ 를 섞음.
  - 그리고 CSP_i 에 send it



3. Prεεch

3. Dummy segment 생성 mechanism

Control knobs

- d: 높으면 privacy ↑, noise injection ↑ (hence, increased monetary cost)
- vocabulary: The size of $\mathcal{V}$ $(m, \Delta, the\ number\ of\ out-of-domain\ words)$

  $\mathcal{V}$ 값이 커지면 privacy ↑, noise size scale $|\mathcal{V}|$ ↑, higher cost
- Voice transformation for noisy segments

  : 음성 복제(Voice cloning), 음성 변환(Voice conversion) 옵션 제공.

  음성 복제(Voice cloning)은 WER 에 영향을 적게 주지만 민감정보(biometric)보호 X,

  음성 변환(Voice conversion)은 WER에 영향을 주지만 민감정보 보호 O
- Number of CSPs used for transcription: employing CSPs 의 수

  여러 개의 CSPs 를 사용하면 금전적 비용을 줄일 수 있지만, 좋은 모델, 나쁜 모델을 함께 사용하므로

  전반적인 효용성은 떨어질 수 있음. (google 이 AWS 보다 성능이 좋았음.)

	Datasets	Google	AWS	Deep Speech
Standard	LibriSpeech	9.14	8.83	9.37
	DAPS	6.70	7.53	10.65
	TIMIT TEST	6.27	7.11	20.08
Non-Standard	VCTK p266	5.15	10.09	26.72
	VCTK p262	4.53	7.87	15.97
	Facebook 1	5.76	7.45	24.72
	Facebook 2	3.07	8.19	26.61
	Facebook 3	8.32	9.42	30.72
	Carpenter 1	9.44	9.44	25.85
	Carpenter 2	9.22	11.53	39.71

Table 1: WER (%) comparison of cloud services, Google and AWS, versus the state-of-the-art offline system, Deep Speech.

3. Preech

4. Voice Conversion

- One-to-One Voice Conversion
  - One-to-one VC maps a predefined source speaker voice to a target speaker voice.
  - ① Original S 및 target speaker's 음성 샘플의 acoustic feature 추출
  - ② Original 과 target 의 feature 를 time-alignment
  - ③ GMM model training
- Many-to-One Voice Conversion
  - Many-to-one VC maps any voice to the same target voice

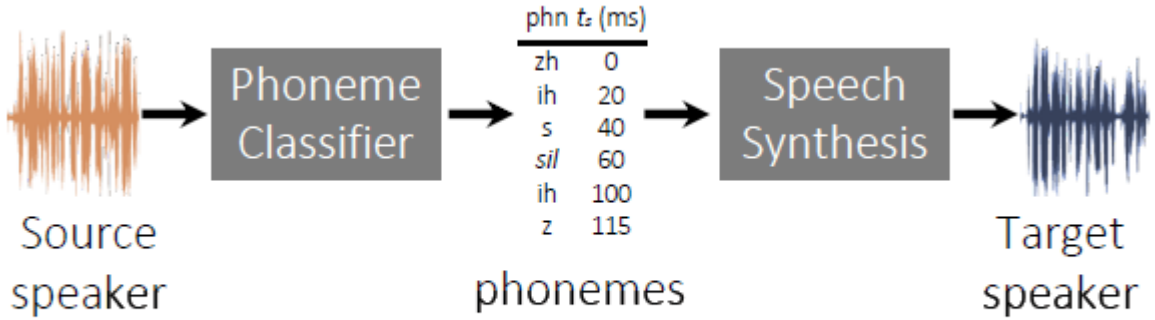


Figure 4: An illustration of the many-to-one VC pipeline.

PPG: Phonetic Posterior grams

① Phoneme Classifier 를 통해 원본 스피커를 PPG로 변환

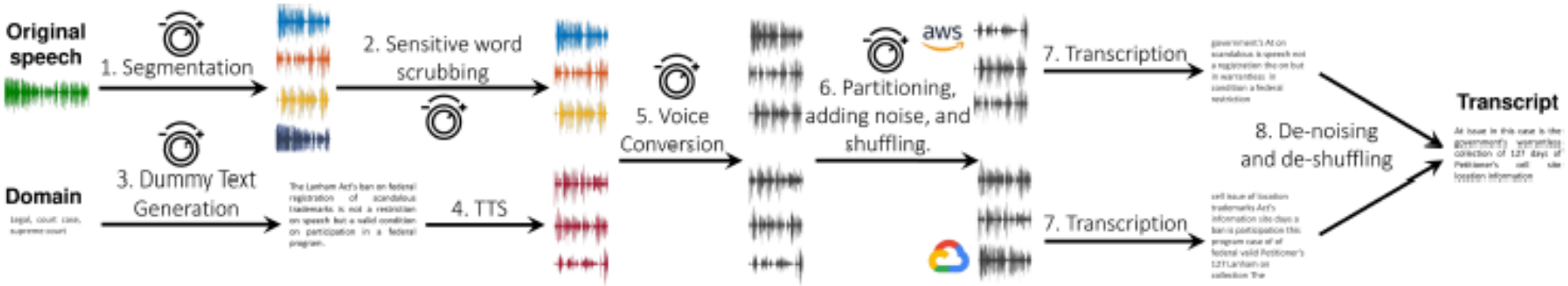
② PPG 를 target speaker 로 변환(합성)

3. Preech

4. Voice Conversion

Control Knobs

- One-to-One Voice Conversion
  - accuracy 높음. 한 사람의 speech 만 학습하기 때문에 변환이 잘됨.
  - 소스와 타깃의 같은 음성 데이터가 매우 많이 필요하고, 등록단계가 필요하여 새로운 화자 추가 어려움.
  - 완벽한 구별 가능성을 제공하지는 않음. (완벽한 privacy 보호 제공 X)
- Many-to-One Voice Conversion
  - One-to-One Voice Conversion 의 단점 보완 가능



4. Evaluation

Q1. Does Prεεch preserve the transcription utility?

프라이버시보호를 적용함으로써 transcription 성능이 떨어지지는 않는가?

Q2. Does Prεεch protect the speaker's voice biometrics?

화자의 성별 등 음성으로 파악가능한 생체 정보를 보호하는가?

Q3. Does Prεεch protect the textual content of the speech?

Text 내용을 보호하는가? (이름, 위치, 의료정보 등 Text로 노출되는 정보)

Q4. Does the different control knobs provide substantial flexibility in the utility-usability-privacy spectrum?

다양한 Control knobs 가 효용성-사용성-프라이버시보호에 유연성을 제공하는가?

4. Evaluation

Q1. Does Prεεch preserve the transcription utility?

프라이버시보호를 적용함으로써 transcription 성능이 떨어지지는 않는가?

Q2. Does Prεεch protect the speaker's voice biometrics?

화자의 성별 등 음성으로 파악가능한 생체 정보를 보호하는가?

Q3. Does Prεεch protect the textual content of the speech?

Text 내용을 보호하는가? (이름, 위치, 의료정보 등 Text로 노출되는 정보)

=〉 Sensitive words scrubbing, DP mechanism 등 다시 설명

Q4. Does the different control knobs provide substantial flexibility in the utility-usability-privacy spectrum?

다양한 Control knobs 가 효용성-사용성-프라이버시보호에 유연성을 제공하는가?

=〉 각 Control knobs 에서 어떻게 제공해주는지 다시 설명

4. Evaluation

Q1. Does Prεεch preserve the transcription utility?

프라이버시보호를 적용함으로써 transcription 성능이 떨어지지는 않는가?

Transcription Utility

	Datasets	Google	AWS	Deep Speech
Standard	LibriSpeech	9.14	8.83	9.37
	DAPS	6.70	7.53	10.65
	TIMIT TEST	6.27	7.11	20.08
Non-Standard	VCTK p266	5.15	10.09	26.72
	VCTK p262	4.53	7.87	15.97
	Facebook 1	5.76	7.45	24.72
	Facebook 2	3.07	8.19	26.61
	Facebook 3	8.32	9.42	30.72
	Carpenter 1	9.44	9.44	25.85
	Carpenter 2	9.22	11.53	39.71

Table 1: WER (%) comparison of cloud services, Google and AWS, versus the state-of-the-art offline system, Deep Speech.

Datasets	Cloning	One-to-One	Many-to-One	OSP
VCTK p266	5.15 (80.73%)	16.55 (38.06%)	21.92 (17.96%)	26.72
VCTK p262	4.53 (71.63%)	7.39 (53.73%)	10.82 (32.25%)	15.97
Facebook1	8.26 (66.59%)	14.60 (40.94%)	20.30 (17.88%)	24.72
Facebook2	9.75 (63.36%)	18.27 (31.34%)	19.44 (26.94%)	26.61
Facebook3	14.93 (51.40%)	23.25 (24.32%)	27.06 (11.91%)	30.72
Carpenter1	14.43 (44.18%)	23.88 (7.62%)	22.63 (12.46%)	25.85
Carpenter2	13.53 (65.93%)	33.71 (15.11%)	38.90 (2.04%)	39.71

Table 2: WER (%) of end-to-end Prεεch which represents the accumulative effect of segmentation, SWS, and different settings of voice privacy and its relative improvement in (%) over OSP (Deep Speech).

4. Evaluation

Q1. Does Pr$\varepsilon\varepsilon$ch preserve the transcription utility?

프라이버시보호를 적용함으로써 transcription 성능이 떨어지지는 않는가?

Transcription Utility

- Many-to-one Voice Conversion 후 CSP transcription 의 성능이 Deepspeech(OSP) 보다 좋음.
- Preech 는 데이터셋마다 다른 성능을 보임. -> VC trainset 의 다양성 부족
  - carpenter 1 의 화자는 크게 말해서 선명하게 들림. -> 성능이 좋음.
  - carpenter 2 의 화자는 작게 말해서 선명하게 들리지 않음 -> 성능이 떨어짐.

4. Evaluation

Q2. Does Prεεch protect the speaker's voice biometrics?

화자의 성별 등 음성으로 파악가능한 생체 정보를 보호하는가?

Voice Biometric Privacy

- Preech 에 VC 적용 후 서로 다른 화자의 speech 를 분리하는 CSP 의 능력 평가
  - multi-speaker datasets 에서 IBM API 는 한 명의 스피커만 존재한다고 하였음.
  - dummy segment (after TTS and VC) 추가한 후 다시 실행 -> 또다시 한 명의 스피커만 존재한다고 감지.

  => 화자의 생체 정보를 숨기고 여러 화자를 단일 화자로 매핑하고, 구별 불가능한 noise 보장

- preech 의 privacy 속성 테스트 (true speaker 와 더 많이 비슷한 adversary 에 대항하는 속성)
  - fake 스피커와 true 스피커의 세그먼트를 Azure 의 스피커 식별 API 에 등록
  - (더미 세그먼트를 추가하고 VC를 적용한 후) Preech 의 세그먼트를 API로 전달
  - Many-to-one VC 가 적용되면 모든 evaluation cases 에서 fake 스피커로 식별.
  - true 스피커라고 분류하지 않음.

  => 실제 화자정보를 숨길 수 있음.

5. Conclusion

Main contributions

1) **End-to-End Practical System**
   - DeepSpeech 에 비해서 preech 가 성능이 더 좋음.
   - Preech 는 음성 생체 인식을 완전히 난독화 함. (개인정보보호 강화)
2) **Non-Standard use of DP(Differential Privacy)**
   - Preech 는 DP를 Non-standard way 로 사용함. 따라서 해결해야할 challenge가 있음.

   (1) noise 가 speech domain 에 추가되어야 한다.

   (2) noise 는 original speech와 구별되지 않아야 한다.
3) **Customizable Design**
   - 사용자가 제어가능한 control knobs 제공

Thank You