Spoken Language Understanding on the Edge

Alaa Saade, Alice Coucke, Alexandre Caulier, Joseph Dureau, Adrien Ball, Théodore Bluche, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibault Lavril, Maël Primet
Snips, Paris
<firstname.lastname>@sonos.com

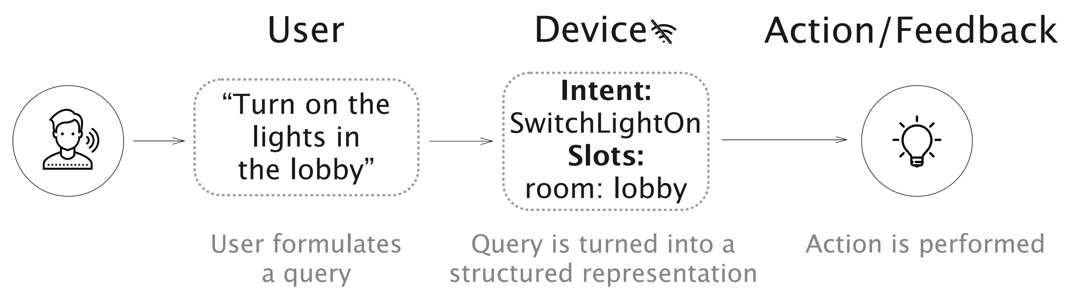
SONOS

1. Contribution

Spoken Language Understanding (**SLU**) is the task of extracting the *intent* and *slots* of a spoken utterance. We present the architecture of a SLU engine that is:

- Cloud-independent and embedded: no remote processing, small enough to run in real time on IoT devices such as the **Raspberry Pi 3** (CPU 1.4GHz, 1GB RAM)
- Private by Design: no user data can be collected
- Accurate: on-par with cloud-based solutions

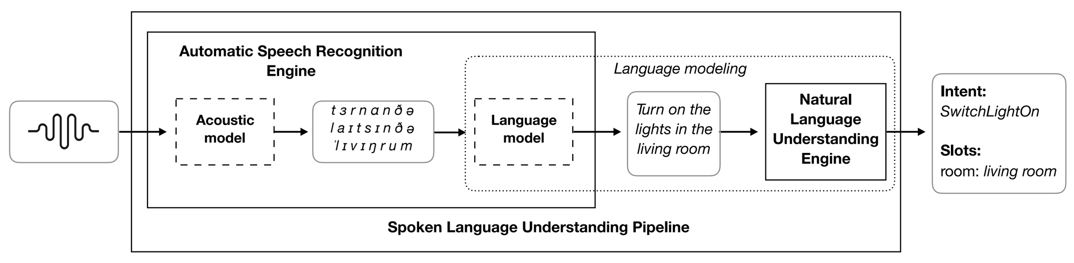
We also release the datasets used in our experiments.



2. Overview

We describe the following components of our SLU ecosystem (see [1] for details):

- an **Automatic Speech Recognition (ASR)** engine, made of
  - an **Acoustic Model (AM)**, mapping raw audio to a phonetic representation
  - a **Language Model (LM)**, turning the prediction of the AM into likely sentences
- a **Natural Language Understanding (NLU)** engine, **open-source** [2], extracting intent and slots from a written query



4. Assistant Contextualization

- Typically the largest component of a SLU engine (up to several TBs in commercial solutions)
- To reduce size and increase accuracy, the LM and NLU are consistent and **contextualized**
- The **same dataset** is used to train both LM and NLU (see example on the right)

5. Language Model

ASR decoding is an approximate best path search in a weighted Finite State Transducer (wFST) decoding graph. This graph is obtained by composing the HCL wFST (mapping the output of the AM to words) with the LM, denoted G (encoding the probability of sequences of words).

Features:

1. accurate and small (1-50MB), **generalizes**
2. the size is furthered reduced by using **dynamic** wFST composition and replacement

Contextualization: G is a class-based wFST LM

$$G = \text{Replace}(G_p, \{G_{s_i}, \forall i \in [1, n]\})$$

- G_p is based on a ngram model trained on patterns in which the slot values are abstracted (`Turn on the light in #ROOM`)
- G_{s_i} models the values of the i-th slot (e.g. a ngram trained on the possible room names)

3. the LM is **customizable privately** (e.g. a list of contacts G_{s_i} can be updated privately)
4. Out-Of-Vocabulary words are detected and discarded through **confidence scoring**

6. NLU

The NLU component performs *intent classification* followed by *slot filling*:

- **intent classification** is based on a logistic regression with BOW features
- **slot filling** relies on a Conditional Random Field model (CRF). One CRF is trained for each intent

NLU and LM are both **contextualized** on the same dataset allowing high-performance on in-domain queries. The NLU is also **customizable privately** consistently with the ngram slot models G_{s_i} of the LM and is already **open source** [2].

7. Results on SmartLights dataset

- End-to-end generalization performance compared with Google's DialogFlow service on a 5-fold cross-validation experiment.
- Metrics are F1-score of Intent Classification and percentage of perfectly parsed utterances (both intent and slots are recovered).
- Assistant total size = 15.1MB.

	Close field		Far field	
Quantity	Snips	Google	Snips	Google
Intent (F1, %)	91.72	89.23	83.56	86.25
Perfect parsing (%)	84.22	79.27	71.67	73.43

8. Results on SmartSpeaker datasets

- English assistant = 65k words, corresponding to 178k pronunciations, 80MB on disk.
- French assistant = 70k words, corresponding to 390k pronunciations, 112MB on disk.
- Metric: percentage of perfectly parsed utterances of the form `Play some music by #ARTIST`.
- The results labeled "Google" correspond to replacing the ASR component by Google's Speech Recognition API. Tier 1 corresponds to artists with popularity rank between 1 and 1,000, tier 2 have ranking between 4,500 and 5,550 and tier 3 between 9,000 and 10,000.

Perfect Parsing (%)		Close field				Far field			
Language	Provider	Tier 1	Tier 2	Tier 3	Average	Tier 1	Tier 2	Tier 3	Average
English	Snips	71.27	67.73	67.21	68.73	42.08	39.36	35.58	39.01
	Google	68.78	37.90	36.74	47.81	58.82	28.85	27.21	38.29
French	Snips	78.20	74.14	73.06	75.13	57.49	53.56	53.89	54.98
	Google	61.04	33.51	32.38	42.31	36.24	15.83	13.47	21.85

3. Acoustic Model

Data: thousands of hours of audio are collected from commercial or public sources. Transcripts are aligned to closely match the audio. The speech corpus is augmented to simulate noisy and far-field conditions.

Architecture and Training: we use a hybrid DNN/HMM model trained with the Kaldi toolkit. The DNN is a TDNN-LSTM network whose hyperparameters are tuned to offer near state-of-the art results while running in real time on a given device. We consider the following architectures (BN: Batch Normalization, pN: projection layer of size N):

Layer Type	Context	nn256	nn512	nn768
TDNN-BN-ReLU	{-2,-1,0, 1, 2}	256	512	768
TDNN-BN-ReLU	{-1, 0, 1}	256	512	768
TDNN-BN-ReLU	{-1, 0, 1}	256	512	768
LSTMP	rec:-3	256, p128	512, p256	768, p256
TDNN-BN-ReLU	{-3, 0, 3}	256	512	768
TDNN-BN-ReLU	{-3, 0, 3}	256	512	768
LSTMP	rec:-3	256, p128	512, p256	768, p256
Num. params		2.6M	8.7M	15.4M

All networks are trained using natural gradient descent and backstitching.

Performance: we compare the Word Error Rate of these models against a state-of-the-art Kaldi recipe (large TDNN) on splits of the Librispeech dataset, in a large vocabulary setting:

Model	dev-c	dev-o	test-c	test-o
nn256	7.3	19.2	7.6	19.6
nn512	6.4	17.1	6.6	17.6
nn768	6.4	16.8	6.6	17.5
KALDI	4.3	11.2	4.8	11.5

The `nn256` AM takes 10MB of memory and runs in real time on the Raspberry Pi 3.

9. Datasets Open-Sourcing

- A **SmartLights** assistant:
  - Language: English
  - Use case: turn on or off the light, change its brightness or color... 6 intents (300 queries / intent)
  - Vocabulary size = 400 words
  - To be used for *cross validation*
- A **SmartSpeaker** assistant:
  - Languages: English and French
  - Use case: 9 playback control intents (volume control track navigation) + play music from large libraries of artists and track
  - Vocabulary size = 65k words
  - To be used for *train* / *test*
  - Test set: 1,500 queries of the form `Play some music by #ARTIST`, where we sample `#ARTIST` from a publicly available list of the most streamed artists on Spotify.
- Audios: close field (0m) and far field (2m)
- Link: `https://research.snips.ai/datasets/spoken-language-understanding`

References

[1] A. Coucke, A. Saade, A. Ball, T. Bluche, A. Caulier, D. Leroy, C. Doumouro, T. Gisselbrecht, F. Caltagirone, T. Lavril, M. Primet, and J. Dureau. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, 2018.

[2] Snips Team. Snips NLU, Snips Python library to extract meaning from text. *GitHub repository*, `https://github.com/snipsco/snips-nlu`, 2018.