Fast ASR-free and almost zero-resource keyword spotting using DTW and CNNs for humanitarian monitoring

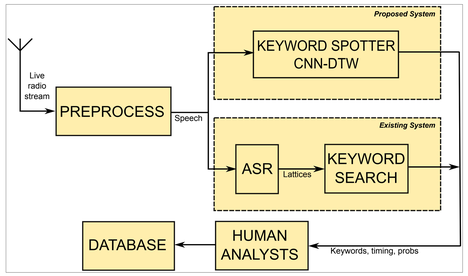
Raghav Menon[1], Herman Kamper[1], John Quinn[2], Thomas Niesler[1]

[1]Stellenbosch University, South Africa [2] UN Global Pulse, Kampala, Uganda

Introduction

- In rural Uganda, where internet connectivity is poor, phone-in talk shows hosted by small community radio stations are used by many to voice problems and concerns.
- The United Nations (UN) has piloted a radio browsing system which uses speech recognition (ASR) and keyword spotting to monitor such talk shows as a means of informing relief and development programmes.
- Availability of transcribed data, even small amounts, has proved to be a key impediment for the development of these radio browsing systems.
- Hence we focus on the development of a keyword spotter that can be set up using resources that are even easier to obtain: a small set of isolated spoken keywords and a larger untranscribed speech corpus.

Radio Browsing System Configuration



Existing and proposed radio browsing system.

- In the existing system, the incoming signals are pre-processed and are then passed on to the ASR which generates lattices.
- The keyword spotting system searches these lattices for the keyword of interest.
- The presence of human analysts allow us to tolerate high false positive rates because false detections can be discarded.
- In the proposed system, we replace the ASR with our proposed CNN-DTW system.

Data

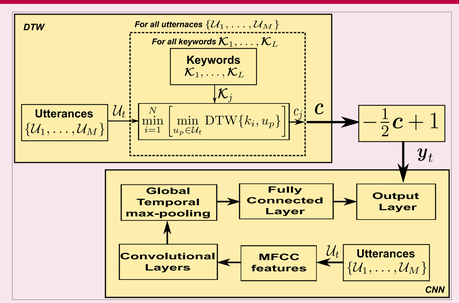
- **Untranscribed in-domain data:** 23 hrs South African Broadcast News (SABN) used for experimental analysis.
  - Test utterances drawn from this corpus.
  - Remainder used as untranscribed speech
  - SABN transcriptions used only for evaluation purposes.
- **Transcribed in-domain data:** Recorded isolated utterances of 40 keywords, each spoken twice by 24 speakers (12 male and 12 female), leading to 1920 labeled isolated keywords.

	Utterances	Speech (h)
Train	5231	7.94
Dev	2988	5.37
Test	5226	10.33
Total	13445	23.64

Keyword Spotting

- Three approaches
  1. ***Dynamic time warping (DTW) based keyword spotting***
     - DTW cost is determined by aligning each keyword with each test utterance within a sliding window.
     - Use cosine cost function and 3 frames skip while sliding over the test utterance.
  2. ***Convolutional neural networks (CNN) keyword spotting***
     - CNN is trained using only 1920 recorded keywords in a supervised manner.
     - 60-frame sliding window applied to test utterance; keyword presence postulated using threshold.
  3. ***CNN-DTW keyword spotting***
     - CNNs require large amount of training data, but are computationally efficient to apply.
     - DTW-based methods can be applied with few keyword exemplars but computationally costly.
     - Hence DTW is used to obtain similarity scores between the small set of isolated keywords and a much larger dataset of untranscribed speech.
     - These similarity scores are used as targets to train the CNN.

Methodology



CNN-DTW keyword spotter training.

- Consider a keyword type $\mathcal{K}$ of which we have N repetitions:

$$\mathcal{K} = (k_1, \ldots, k_i, \ldots, k_N)$$

  where each k_i is the sequence of speech features for the i^{th} exemplar of keyword $\mathcal{K}$.
- To obtain the DTW-based score indicating how likely it is that an utterance $\mathcal{U}$ contains an instance of keyword $\mathcal{K}$, calculate:

$$c = \min_{i \in 1 \ldots N} \left[\min_{u_p \in \mathcal{U}} \text{DTW}\{k_i, u_p\} \right]$$

- Each u_p is a successive segment of utterance $\mathcal{U}$, and $\text{DTW}\{k_i, u_p\}$ is the DTW alignment cost between the speech features of exemplar k_i and the segment u_p.
- Thus, we determine relevance of keyword using the lowest cost encountered when sweeping each exemplar over the utterance.
- We calculate the cost c separately for each of the L keyword types.
- For utterance $\mathcal{U}$ obtain costs $[c_1, \ldots, c_j, \ldots c_L]$ where $c_j \in [0, 2]$.
- Normalization $y_j = -\frac{1}{2}c_j + 1$ applied to ensure $y_j \in [0, 1]$, with 1 indicating a perfect match and 0 indicating maximum dissimilarity.
- Train a CNN to predict this target vector with $\mathcal{U}$ as input.

Experimental setup

- Three baseline systems were used for evaluation.
  1. **DTW-QbyE** where DTW is performed for each exemplar keyword on each utterance, and the resulting scores are averaged.
  2. **DTW-KS** where the minimum (best) score over all exemplars of a keyword type is used per utterance.
  3. **CNN** as an end-to-end CNN classifier trained only on the isolated words.
- Performance is reported in terms of
  1. **AUC** which indicates the performance of the model independent of a threshold, with higher AUC indicating a better model.
  2. **EER** The point at which the false positive rate equals the false negative rate (lower is better).

Results and Discussion

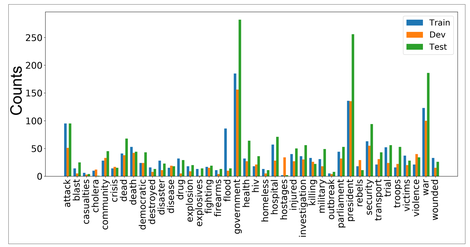
Keyword spotting performance and execution time on the test set in minutes.

	AUC		EER		Time
	dev	*test*	*dev*	*test*	(min)
CNN	0.5698	0.5448	0.4435	0.4771	55
DTW-QbyE	0.6639	0.6612	0.3864	0.3885	900
DTW-KS	0.7556	0.7515	0.3092	0.3162	900
CNN-DTW	0.636	0.6285	0.4073	0.4161	5
CNN-DTW with GNL	0.6443	0.6357	0.4036	0.4092	5

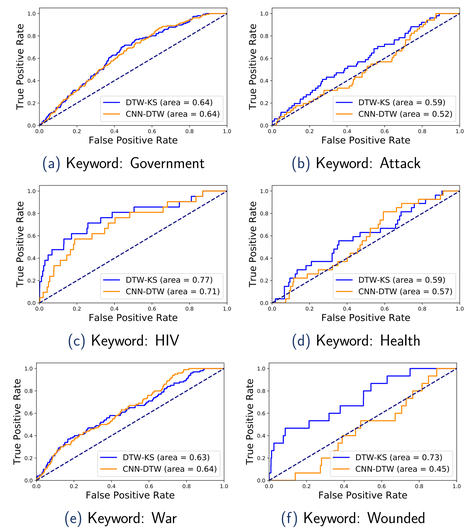
- The DTW-KS system achieves the best result.
- Combining CNN and DTW performs similarly to DTW-QbyE.
- CNN end-to-end keyword spotting provides the worst result.

Qualitative Analysis of the 3 best performing and the 3 worst performing keywords. The number of occurrences of each keyword in the SABN corpus is shown in brackets. The absolute number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) are shown.

Government (156)		**Attack (51)**	
TP: 93	FP: 1149	TP: 25	FP: 1481
FN: 63	TN: 1683	FN: 26	TN: 1456
(a)		(b)	
HIV (21)		**Health (27)**	
TP: 14	FP: 1032	TP: 14	FP: 1422
FN: 7	TN: 1935	FN: 13	TN: 1539
(c)		(d)	
War (100)		**Wounded (15)**	
TP: 58	FP: 1222	TP: 8	FP: 1452
FN: 42	TN: 1666	FN: 7	TN: 1521
(e)		(f)	



Keyword occurrence distribution in the SABN corpus.



(a) Keyword: Government (b) Keyword: Attack

(c) Keyword: HIV (d) Keyword: Health

(e) Keyword: War (f) Keyword: Wounded

Receiver operating characteristics for the selected keywords.

- The performance of the proposed system is closer to the DTW baseline for more frequent keywords.
- The DTW baseline took approximately 15 hours to process all 40 keywords and all utterances in the 10-hour test set on a 20-core machine.
- The CNN-DTW system processed the same data in approximately 5 minutes on a PC with a single GeForce GTX 1080 GPU.

Conclusion

- By combining CNN and DTW, it is possible to obtain a ASR-free keyword spotting system which is fast enough for real-time processing.
- The performance of the CNN-DTW system is comparable to DTW-QbyE, giving an AUC of 0.64, but much more computationally efficient.
- It was assumed that almost all of the available data was untranscribed and only a small number of isolated keywords are required. Hence CNN-DTW is suitable for low-resource keyword detection.