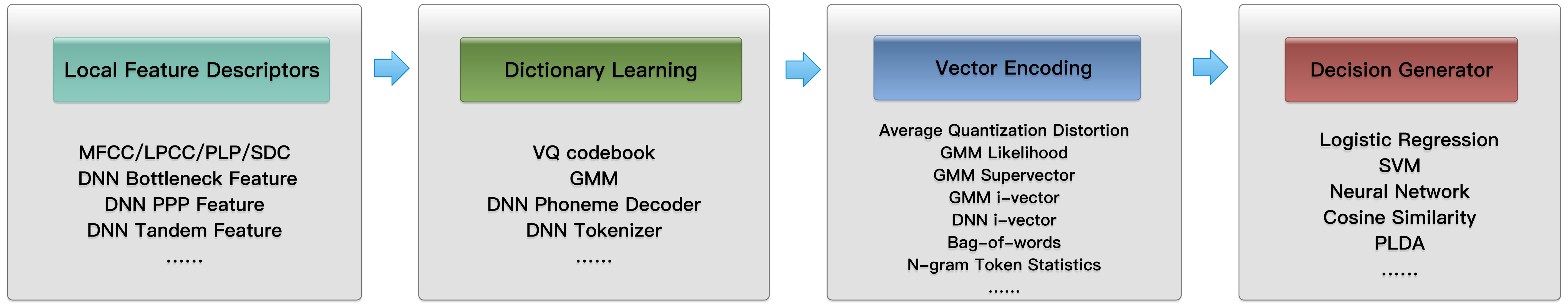
Insights into End-to-End Learning Scheme for Language Identification

Weicheng Cai[1], *Zexin Cai*[1], *Wenbo Liu*[3], *Xiaoqi Wang*[4] and *Ming Li*[1,2]

1. School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China
2. Data Science Research Center, Duke Kunshan University, Kunshan, China
3. Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, USA
4. Jiangsu Jinling Science and Technology Group Limited, Nanjing, China

ml442@duke.edu

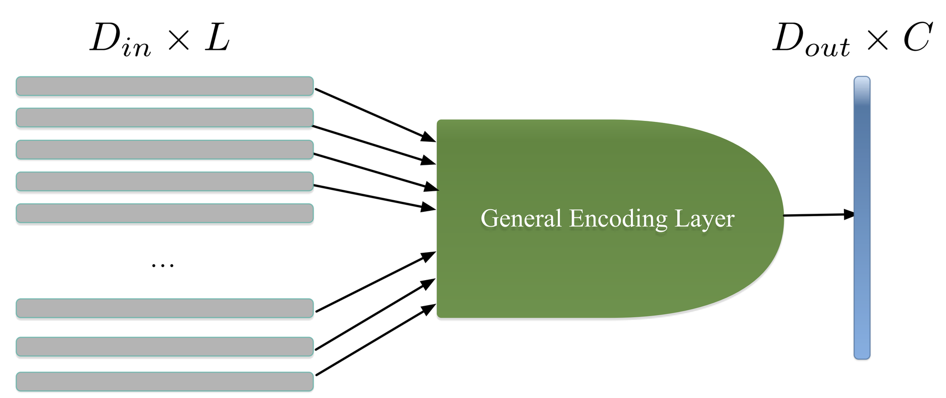
Introduction



Four main steps in the conventional processing pipeline

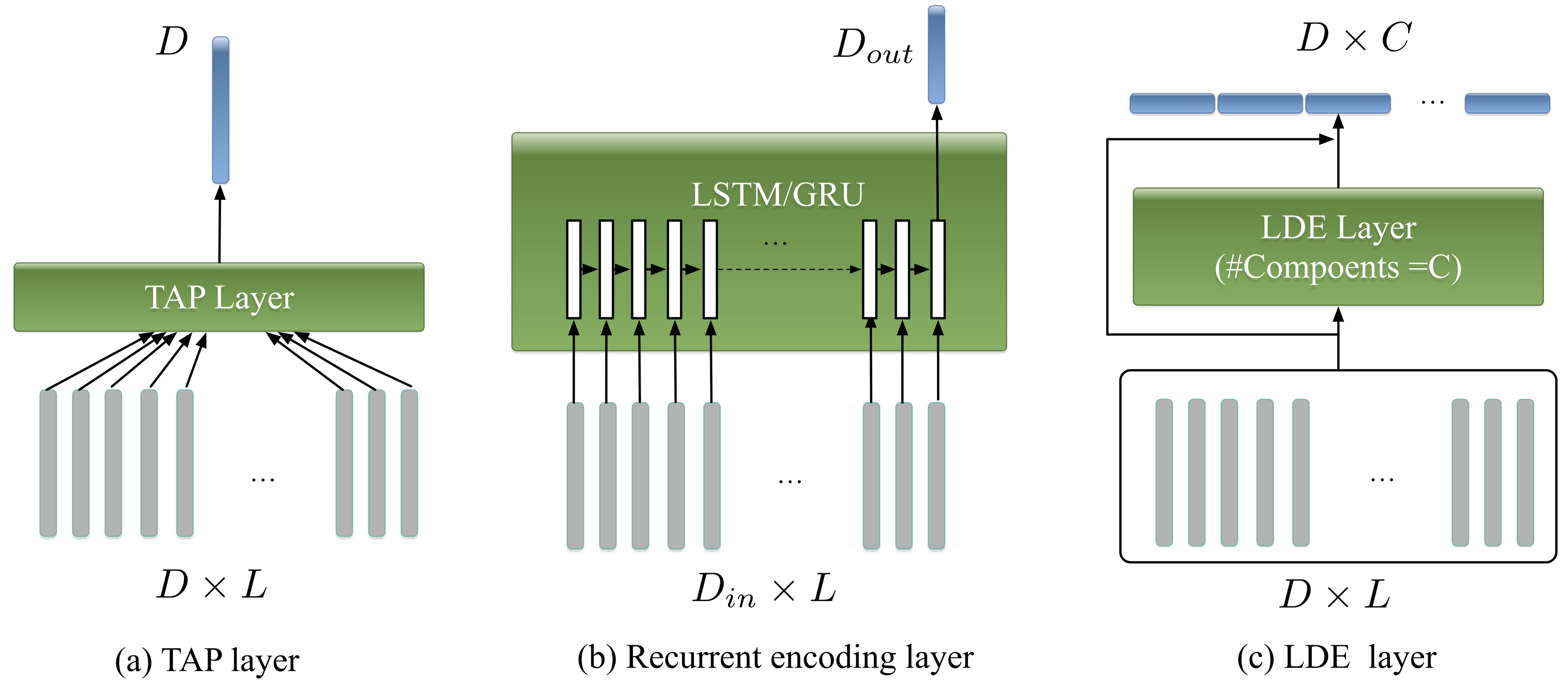
The GMM i-vector based approaches are comprised of a series hand-crafted or ad-hoc algorithmic components, and they show strong generalization ability and robustness when data and computational resource are limited.

General Encoding Layer



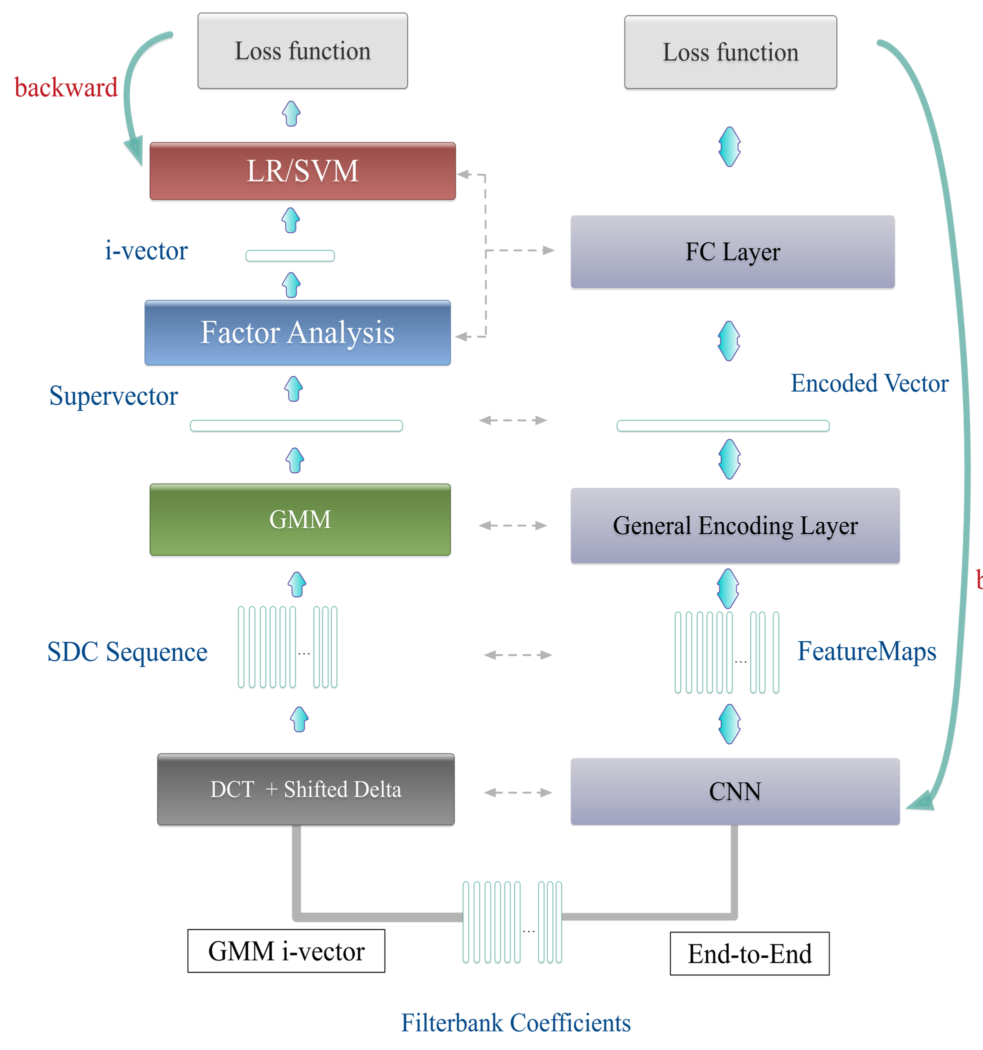
The encoding layer plays a role in extracting a fixed-dimensional utterance level representation from a variable-length input sequence.

- **Temporal average pooling (TAP) layer**
- **Recurrent encoding layer**
- **Learnable dictionary encoding (LDE)layer**

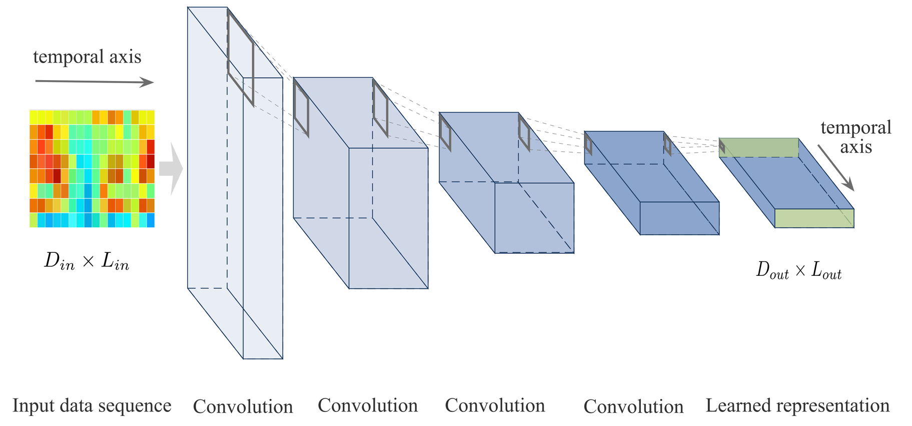


(a) TAP layer (b) Recurrent encoding layer (c) LDE layer

End-to-End Learning Scheme



- **In traditional GMM i-vector method, each component is optimized in separate steps. In our end-to-end learning scheme, the entire pipeline is learned in an integrated manner because the features, encoding layer and the encoded vector representation for the classifier are all learned jointly.**
- **There are two characteristics of our learning scheme showing the theoretical and practical compatibility with the classical GMM i-vector approach:**
  - **Each component of our neural network has its parallel equivalent block towards to the classical GMM i-vector processing stream.**
  - **Our neural network architectures accept variable-length speech inputs and give an utterance level result.**



- The role of CNN
  - **The convolution layer of CNNs operates in an sliding window manner acting as a automatic local feature extractor.**
  - **Since anywhere outside the receptive field of a unit does not affect the value of that unit, it is necessary to carefully control the receptive field, to ensure that it covers the entire relevant input region.**

Experimental Results and Discussion

Table 1. Performance on the 2007 NIST LRE closed-set task

System ID	System Description	$C_{avg}(\%)/EER(\%)$ 3s	10s	30s
1	GMM i-vector	20.46/17.71	8.29/7.00	3.02/2.27
2	DNN i-vector	14.64/12.04	6.20/3.74	2.601.29
3	DNN PPP Feature	**8.00/6.90**	**2.20/1.43**	**0.61/0.32**
4	DNN Tandem Feature	9.85/7.96	3.161.95	0.97/0.51
5	DNN Phonotactic[22]	18.59/12.79	6.28/4.21	1.34/0.79
6	RNN D&C[22]	22.67/15.57	9.45/6.81	3.28/3.25
7	LSTM-Attention[21]	-/14.72	-/-	-/-
8	**CNN-TAP**	9.98/11.28	3.24/5.76	1.73/3.96
9	**CNN-GRU**	11.31/10.74	5.49/6.40	-/-
10	**CNN-LSTM**	10.17/9.80	4.66/4.26	-/-
11	**CNN-LDE**	**8.25/7.75**	**2.61/2.31**	**1.13/0.96**

- For ID2 to ID5, additional speech data with transcription and an extra DNN phoneme decoder is required, while our end-to-end systems only rely on the acoustic level feature of LID data.
- For each training step, an integer L within [200,1000] interval is randomly generated, and each data in the mini-batch is cropped or extended to L frames. In testing stage, all the 3s, 10s, and 30s duration data is tested on the same model. Because the duration length is arbitrary, we feed the testing speech utterance to the trained neural network one by one.
- It's very interesting that although recurrent layer introduces much more parameters comparing with TAP, it results in a slightly degraded performance. Specially, when the full 30s duration utterance is fed into our CNN-GRU/CNN-LSTM neural network trained within 1000 frames (10s), it suffers from "the curse of sentence length". The performance drops sharply and almost equals to random guess.

- **Although recurrent layer can deal with variable-length inputs theoretically, it might be not suitable for the testing task with wide duration range and particularly with duration that are much longer than those used for training.**
- **The success of TAP and LDE layer inspires us that it might be more necessary to get utterance level representation describing the context-independent feature distribution rather than the temporal structure.**