CSLT TECHNICAL REPORT-20160030 [Saturday 7th January, 2017]

Collaborative Joint Training with Multi-task Recurrent Model for Speech and Speaker Recognition

Zhiyuan Tang[1,2]
, Lantian Li[1,2]
, Dong Wang[1,3*]
and Ravichander Vipperla[4]

*Correspondence: wang-dong99@mails.tsinghua.edu.cn
[1]Center for Speech and Language Technology, Research Institute of Information Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China
Full list of author information is available at the end of the article

Abstract

Automatic speech and speaker recognition are traditionally treated as two independent tasks and are studied separately. The human brain in contrast deciphers the linguistic content and the speaker traits from the speech in a collaborative manner. This key observation motivates the work presented in this article. A collaborative joint training approach based on multi-task recurrent neural network models is proposed, where the output of one task is back-propagated to the other tasks. This is a general framework for learning collaborative tasks, and fits well with the goal of joint learning of automatic speech and speaker recognition. Through a comprehensive study, it is shown that the multi-task recurrent neural net models deliver improved performance on both automatic speech and speaker recognition tasks as compared to single-task systems. The strength of such multi-task collaborative learning is analyzed and the impact of various training configurations is investigated.

Keywords: Speech Recognition; Speaker Recognition; Recurrent Neural Networks; Multi-task Learning; Joint training

1 Introduction

Automatic speech recognition (ASR) and speaker recognition (SRE) are two important tasks in speech processing. Traditionally, these two tasks are treated independently and are studied separately. This leads to task-specific learning methods and models. However, this independent treatment is not the way that we humans process speech signals: we always simultaneously decipher speech content and other meta information including languages, speaker characteristics, emotions etc. This 'multi-task processing' relies on two premises: (1) all these tasks share the same signal processing pipeline in our aural system, and (2) they are mutual beneficial, i.e., the success on one task improves the performance on others. This observation has motivated us to consider the possibility to deal with multiple tasks with a unified model. In this paper, we focus on speech and speaker recognition, and demonstrate that they can be addressed by a single neural-net model based on deep learning.

1.1 Speech and speaker recognition: close correlation

The close correlation between ASR and SRE has been recognized for some time. Firstly, many common techniques have been designed and employed for the two tasks, from the use of MFCC (Mel-frequency cepstral coefficients) features to the HMM/GMM (hidden Markov model/Gaussian mixture model) modeling framework. Secondly, research in both areas has benefited by inter-exchange of ideas. For example, the success of deep neural networks (DNN) in ASR [1, 2] has motivated the usage of neural models in SRE [3, 4]. Thirdly, it has been observed that employing knowledge derived from one task for solving the other task is often beneficial. For example, speaker identities (e.g., i-vectors) derived from SRE have been found to improve the ASR accuracy [5, 6], and phone posteriors derived by ASR have been successfully applied to improve SRE performance [7, 8, 9, 10, 11]. BenZeghiba et al. [12] proposed a joint decoding algorithm for ASR and SRE, by searching for phone and speaker identities that maximize their joint probability. All the above studies try to exploit the correlation between ASR and SRE for mutual benefit; however, none of them formulates the idea as a unified model that learns and addresses the two tasks jointly.

The development of deep learning methods in speech processing has opened new research avenues. Since 2011, DNN and its recurrent variant, recurrent neural networks (RNN) have become the new state-of-the-art for ASR [13, 14]. Recently, the same model has also achieved outstanding results in SRE, at least in text-dependent tasks [15]. The deep RNN structure has two main influences: first, the structural depth (multiple layers) produces high-level features that are more task-oriented; second, the temporal depth (recurrent connections) allows learning complex temporal patterns. Both the modern ASR and SRE systems leverage this to their advantage and achieve their respective state-of-the-art performances. A natural question that arises is: Given that both the two tasks use deep RNN models, would it be possible to merge the two RNN models into a single one that can learn ASR and SRE jointly and infer the two tasks simultaneously, as we human do all the time?

1.2 Difficulty with information competition

The 'multi-task learning' [16] makes the joint training and inference for correlated tasks possible. The basic idea of this learning approach is that if two tasks are correlated, then part of their model structures can be shared. This structure sharing allows the two tasks to share the data statistics, leading to more robust models. A typical multi-task learning strategy in the deep learning framework is to share the low-level layers of neural nets, while keeping the higher-level layers task-dependent. This is essentially a feature sharing strategy [17].

This multi-task learning approach based on structure sharing has been widely used in multilingual speech recognition, where the ASR tasks on different languages are treated as correlated, and the feature extraction component can be shared, due to the commonality of human languages [18, 19, 20]. This approach has been found to be especially useful for low-resource languages for which only limited training data are available [21, 22]. As another example, Chen and colleagues [23] found that phone recognition and grapheme recognition can be treated as two correlated tasks, and a DNN model trained with the two tasks as objectives outperforms the

ones trained only with phone targets. Other multi-task learning research work has been reviewed in [17].

While this structure sharing approach, in particular feature sharing, is simple and effective, it is not readily applicable to joint training ASR and SRE. This is because the effective features required for the two tasks are fundamentally different: ASR requires features involving linguistic content as much as possible, whereas SRE requires features with rich speaker information. These two requirements are indeed mutually exclusive: to represent linguistic content, it is desirable to suppress the speaker-related information and vice-versa for the speaker content. Such tasks using different parts of information in the input data are regarded as 'information-competitive' tasks. For these tasks, the conventional multi-task learning methods based on structure/feature sharing are clearly not ideal. Unfortunately, many correlated tasks fall under this category, e.g., language identification and speaker recognition, emotion recognition and speech recognition. Multi-task joint learning for such information-competitive tasks has not been fully investigated.

1.3 Our proposal

A key observation for information-competitive tasks is that they are 'collaborative', which means that their performance can be boosted by leveraging information from each other. This collaborative relation among tasks can be leveraged to design a new multi-task joint learning framework. This framework is not based on structure sharing; instead it relies on inter-task information propagation, with which the performance of each task can be boosted by the auxiliary information derived from other tasks. We call this learning framework for collaborative tasks 'collaborative joint learning', or simply 'collaborative learning'. This learning framework is generic and as such can be applied to any collaborative tasks as long as they are complementary, for instance ASR and SRE.

This paper presents such a collaborative learning structure based on deep neural net models. The key idea is to merge task-specific neural models with inter-task recurrent connections into a unified model. This model fits well the ASR/SRE joint training. In this scenario, the speech content and speaker identity are produced at each frame step by the ASR and SRE components respectively. By exchanging these bits of information, performances on both ASR and SRE are sought to be improved. This leads to a joint and simultaneous learning for the two tasks, simulating the process of language acquisition in human beings (c.f. Section 2).

A preliminary version of this paper has been published in arXiv [24]. This paper extends [24] with several contributions including: (1) presentation of more evidence from cognitive studies on human auditory systems to support the collaborative joint learning idea; (2) detailed analysis of the modeling strength of the collaborative learning and the multi-task recurrent neural net model; (3) extension of the experimental work from fully-labeled training to partially-labeled training, where the training data are labeled with targets of one partial task; (4) presentation of more comprehensive experimental results to show how ASR and SRE impact each other in the collaborative learning framework.

Note that Li et al. [25] proposed a similar model structure with primary focus on ASR with SRE being treated as an auxiliary task. We demonstrate that the

recurrent structure is more generic than providing auxiliary information; it is a way of collaborative learning that can jointly improve the performance of all the constituent collaborative tasks.

The rest of the paper is organized as follows: Section 2 presents some cognitive evidence for speech and speaker collaborative learning, and Section 3 presents the learning framework based on multi-task recurrent models and analyzes the strength of this learning approach. Section 4 describes details of the ASR/SRE multi-task recurrent model, and Section 5 reports the experimental results. The paper is concluded in Section 6, with some ideas for future work.

2 Cognitive evidence

The strong connection between speech and speaker perception has been found in numerous cognitive studies from infant language development to audio perception in adults.

Research on information disentanglement in language development of infants show that in early developmental stages, infants can not distinguish linguistic content from other factors e.g., gender and emotion. They gradually learn which information is relevant for each task, thus enabling them to distinguish latent factors such as semantic meaning and speaker identity. Walker et al. [26] reported that infants are sensitive to gender information in their early state of language development. Houston and colleagues [27, 28] found that infants at 7.5 months treat the same word from male speakers and female speakers as different words, while infants at 10.5 months could generalize different instances of the same word across utterances of the opposite sex.

The strong influence of speech and speaker perception on each other is highlighted in a study by Johnson [29], where it was found that 7-month-old infants could detect speaker change when the pronounced sentences were familiar to them. If the words were in another language or the sentences were read in reverse, they could not detect the speaker change.

According to the PRIMIR framework presented by Werker and colleagues [30], understanding performance changes of an infant in any one task requires consideration of performance in other tasks, though the inter-task mutual impact is different at different times in development.

All the above research, and in particular the PRIMIR theory, provides strong support for our study on multi-task collaborative learning. The network structure shown in Fig. 1 (elaborated in the next section) is like the auditory system of an infant: the shared acoustic pre-processing component corresponds to the 'bias filter' in PRIMIR and plays the role of removing irrelevant information like background noise; the single-task network is the 'development structure' that can pick up task-dependent information for specific tasks; the multi-task recurrence is the 'inter-task co-development structure' that leverages correlations among different language tasks.

Multi-task information disentanglement is not only an important process for infants when developing their language, but also an important tool that adults use everyday to process various perceptual tasks. For example, Creelman et al. [31] found that word recognition accuracy decreased in the presence of noise when the

identity of the talker was unpredictable from trial to trial. The same phenomenon was also found by Summerfield [32], Verbrugge et al. [33], Mullennix et al. [34] and Nusbaum et al. [35]. All these studies conclude that speaker variation impacts word recognition in a significant way. Johnson [36] summarized these studies and described a 'talker normalization' framework, which argues that a listener needs to locate a talker in a 'talker coordinate' to assist the recognition of the following word. The coordinate can be established very quickly at the very beginning of the listening task, by estimating the talker's vocal track length or retrieving words in his/her memory that are similar to the talker's words.

Another interesting study is the work conducted by Eklund and colleagues [37]. They found evidence of a connection between talker perception and vowel perception, this time in a study of whispered speech. When listeners misidentified the sex of the talker their vowel identification error rate was 25%, but when they correctly identified the sex of the speaker the vowel error rate was only 5%. This suggests that talker perception and vowel perception are interconnected with each other. The argument of the connection between word recognition and speaker identification was clarified by Nygaard [38]. It was clearly demonstrated that linguistic and non-linguistic properties are integrally related components of the same acoustic speech signal, and consequently, the speech perception process.

The above studies suggest that the inter-link between speech and speaker recognition tasks not only plays a role in infant language development, but also in adult's daily perceptual tasks. This provides further support for our research, that multiple correlated tasks should not only be considered in model training, but also be considered during inference. The multi-task recurrent architecture that will be elaborated in the next section provides a framework to train and infer various tasks jointly in a principled way.

3 Collaborative joint learning

The cognitive evidence has motivated us to consider a joint training architecture where collaborative tasks can exchange information with each other so that performance of the individual tasks can be simultaneously improved. This section first presents a joint learning framework for collaborative tasks based on an inter-task recurrent neural network structure, and then analyzes the strength of the proposed model. Finally we compare the recurrent model and the feature sharing approach, highlighting their respective advantages and application scenarios.

3.1 Multi-task recurrent model

An ideal model for collaborative learning should at least possess the following two properties:

- it should be a unified model where the components for individual tasks are homogeneous in model structure and learning scheme, so that they can be trained jointly and simultaneously as a single model;
- the components for individual tasks should be able to collaborate with each other, i.e., they should help each other in both training and inference.

A multitude of model structures satisfy these requirements, e.g., joint factor analysis (JFA) has been widely used to model speaker traits and channel effect simultaneously [39]. In this paper, we are most interested in deep neural models due to

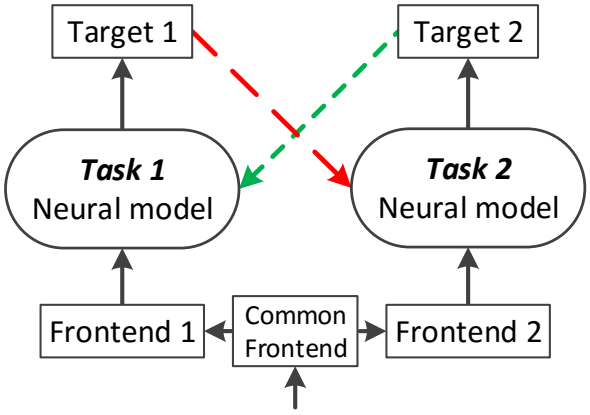


Figure 1 *The multi-task recurrent neural model for collaborative learning.*

their superior performance in speech and speaker recognition. Motivated by the cognitive evidence, a possible collaborative learning structure with deep neural models is illustrated in Fig. 1, where each individual task is modeled by a deep neural model, and some inter-task connections are introduced to propagate information across tasks.

This structure perfectly meets the requirements for collaborative learning: it is a unified model purely based on neural networks and is therefore homogeneous, and the individual components help each other by the recurrent information exchange. Note that the inter-task connections propagate information from output back to input, so the entire structure is an RNN model – the only difference from the vanilla RNN model is that the recurrent connections are at the task level. We denote this model as a multi-task recurrent model, and each task-specific neural structure as a 'component'. The information propagated back is called as 'feedback information', and is referred as 'auxiliary information' when fed into a component.

3.2 Strength of multi-task recurrent model

The strength of the multi-task recurrent model can be understood in different ways. We focus on two perspectives: for task specific component the model offers context-aware learning; when viewed in its entirety, the model provides a mechanism for task-specific information disentanglement in the supervised learning paradigm

3.2.1 Context-aware learning

Context-aware learning involves extra information when training a neural model. For example, [5] introduced a speaker vector (i-vector) as additional input to improve DNN-based ASR, and [40] showed that incorporating the rate of speaking improved their ASR system. Intuitively, the extra information can provide more cues to discriminate the targets; more formally, involving the extra information leads to a context-aware conditional model that is easier to train.

Consider a particular component in the recurrent structure, e.g., the ASR component. Let x and t denote the primary input features (e.g., Fbank) and the targets (e.g., phones) respectively, and c be the extra input obtained from other components

(e.g., speaker vector). With the information c, the model estimates the probability $P(t|x,c)$. If we regard the extra input c as a context indicator, the model is context-aware. Note that the context-aware model is a conditional model with the context c as the condition. In contrast, the single-task model, which can be formulated as $P(t|x)$, is essentially a marginal model $\sum_c P(t|x,c)P(c|x)$ where c is latent.

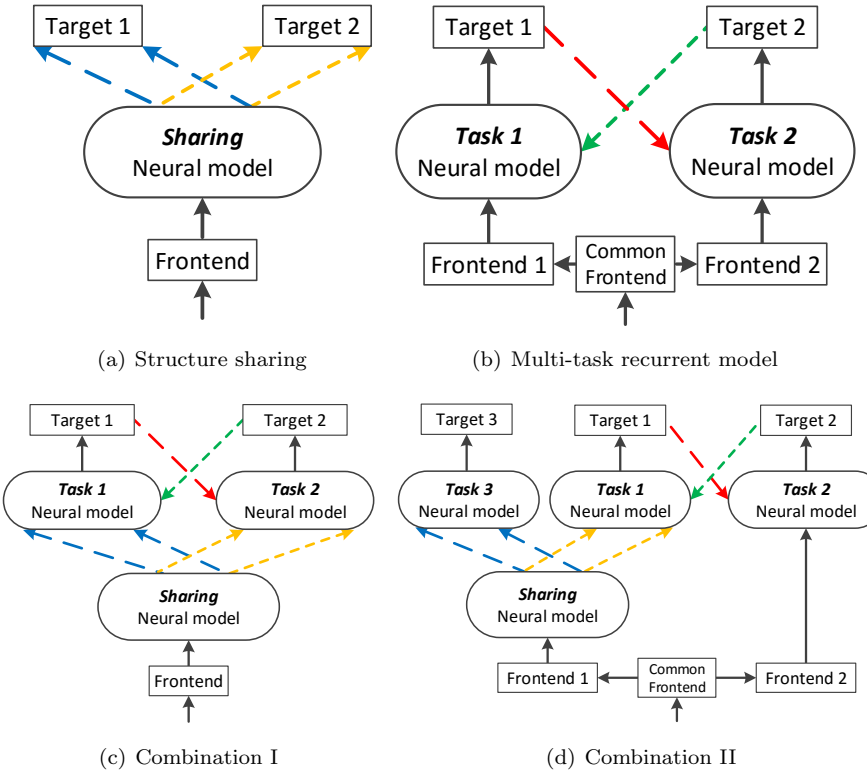
The shift from a marginal model to a conditional model offers at least two advantages: firstly, the context probability $P(c|x)$ is learned by a separate model, which is not only more flexible but also more effective, as more powerful models can be used. Secondly, the conditional model is often easier to train than the marginal model, since the latter tends to involve a more uneven cost function (with more modes), due to the effect of the latent variable c.

3.2.2 Supervised information disentanglement

Another strength of the multi-task recurrent model is its capability of disentangling task-sensitive information. Information entanglement is a major challenge in most machine learning tasks. Taking ASR and SRE as an example, the causal factors for the two tasks (linguistic contents and speaker traits) are mixed together and are buried in raw signals. Extracting the most effective information (factors) for each task is one of the most challenging subjects in both ASR and SRE research. For a long time, researchers focused on designing elegant filters and transforms to obtain task-sensitive features and suppress irrelevant ones. Recently, deep neural models have been widely used to learn task-sensitive features through a layer-by-layer structure [41].

A successful approach to the layer-wise feature learning is through unsupervised learning methods such as stacked RBM [42] or stacked autoencoder [43]. One of the advantages of the unsupervised learning approach is that it can disentangle prominent factors within the signal [41]. This has been used to understand the success of the unsupervised pre-training methods, where the pre-trained models are used to disentangle information, by which the task-related features can be easily extracted by simple fine tuning [41, 44]. For supervised learning of DNN models, this information disentanglement is however not so obvious. Instead of disentangling information and then selecting the relevant features, supervised learning is more like a layer-by-layer information filtering: it chooses a particular task, and then tries to select features that are most relevant to the task and suppress irrelevant ones. A shortcoming of this process is that there is only one task to supervise the information filtering, which is sometimes not very effective.

With the collaborative learning using multi-task recurrent model, tasks collaborate together to discover their own desired feature. Specifically, each task informs others which information it prefers and which information it does not, which can help other tasks in feature extraction. For example, if component A informs component B that it wants some piece of information, and component B informs component A that it does not want the information, there is a strong evidence that the information is closely related to task A. This collaborative information extraction process is essentially an information disentanglement procedure, and is within the supervised learning paradigm. We conjecture that this information disentanglement leads to more precise feature extraction for each individual task compared to the single-task systems, as the latter is supervised by and oriented to a single task.



(a) Structure sharing (b) Multi-task recurrent model

(c) Combination I (d) Combination II

Figure 2 *Various multi-task joint learning models.*

3.3 Comparison of two joint learning models

The structure sharing approach and the multi-task recurrent model are both effective methods for multi-task joint learning, though with very different rationalities. For a clear comparison, the two approaches are illustrated in Fig 2, plot (a) and (b) respectively. Several important differences are listed as follows.

- The rationality of structure sharing is to accumulate statistical strength from individual tasks so that each model can be trained more robustly. The rationality of the multi-task recurrent model, however, lies in borrowing information from each other so that more accurate models for individual tasks can be learned. Analogous to 'structure sharing', this can be regarded as 'information sharing'.
- The structure sharing approach tends to be more effective for tasks with similar targets. For example, in multilingual ASR, the targets of all the individual components are phone discrimination. This target similarity enables sharing in feature extraction and in model sub-structures. In contrast, the multi-task recurrent model is more effective for tasks with heterogeneous targets, for which information from each of them is collaborative.
- The structure sharing approach focuses only on model training – once the models have been trained, the task specific models are often used independently. The multi-task recurrent model, in contrast is a unified model, so the collaborative tasks must be learnt as well as inferred simultaneously.

- Due to the accumulation of sufficient statistics across tasks, the structure sharing approach is particularly useful in scenarios with data sparsity, making it a good tool in transfer learning (e.g., in minor language ASR). For the multi-task recurrent model, however, all individual components should be well trained with sufficient data, otherwise the joint model may perform poorly.
- Due to the interaction among individual components, multi-task recurrent models are generally more difficult to train as compared to models with shared structure.

The differences listed above are not absolute. For instance, structure sharing can be also used for tasks with heterogeneous targets, in which case only one task is of primary concern and others are auxiliary. Moreover, the two approaches can be combined to construct more complex multi-task learning models, e.g., the models shown in Fig. 2 (c) and (d). This paper focuses on the typical multi-task recurrent model shown in Fig. 2 (b), and leaves more complex variants for future exploration.

4 Multi-task recurrent model for ASR and SRE

Applying the multi-task recurrent model for ASR and SRE is straightforward. We choose the structure illustrated in Fig. 1, where the two neural models in the diagram correspond to the ASR and SRE components respectively. We first describe the single-task baseline model used in our study, and then present the multi-task recurrent model.

4.1 Basic single-task model

The state-of-the-art architecture for ASR is based on RNN, in particular the long short-term memory (LSTM) model [13]. This model has also delivered good performance on SRE task [15]. We therefore choose LSTM to build the single-task baseline systems for both ASR and SRE. The modified LSTM structure proposed in [14] is used. The network structure is shown in Fig. 3.

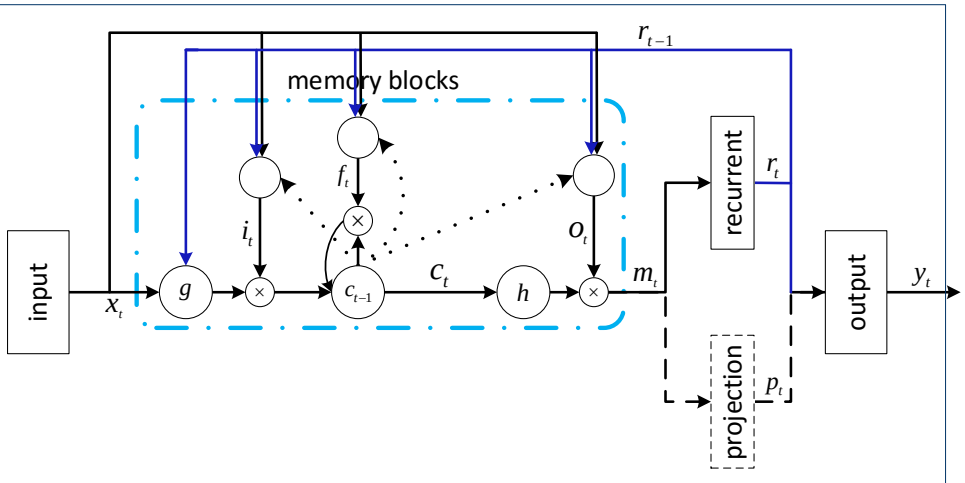


Figure 3 *The LSTM model for ASR and SRE single-task baselines. The picture is reproduced from [14].*

The associated computations are as follows:

$$
\begin{aligned}
i_t &= \sigma(W_{ix}x_t + W_{ir}r_{t-1} + W_{ic}c_{t-1} + b_i) \\
f_t &= \sigma(W_{fx}x_t + W_{fr}r_{t-1} + W_{fc}c_{t-1} + b_f) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cr}r_{t-1} + b_c) \\
o_t &= \sigma(W_{ox}x_t + W_{or}r_{t-1} + W_{oc}c_t + b_o) \\
m_t &= o_t \odot h(c_t) \\
r_t &= W_{rm}m_t \\
p_t &= W_{pm}m_t \\
y_t &= W_{yr}r_t + W_{yp}p_t + b_y
\end{aligned}
$$

In the above equations, the W terms denote weight matrices and those associated with the cells were constrained to be diagonal in our implementation. The b terms denote bias vectors. x_t and y_t are the input and output symbols respectively; i_t, f_t, o_t represent respectively the input, forget and output gates; c_t is the cell and m_t is the cell output. r_t and p_t are two output components derived from m_t, where r_t is recurrent and fed to the next time step, while p_t is not recurrent and contributes to the present output only. $\sigma(\cdot)$ is the logistic sigmoid function, and $g(\cdot)$ and $h(\cdot)$ are non-linear activation functions, often chosen to be hyperbolic. $\odot$ denotes element-wise multiplication.

4.2 Multi-task recurrent model

We use the recurrent LSTM model used in the single-task systems to build the ASR component and the SRE component, and introduce inter-task recurrent connections to construct the multi-task recurrent model. Note that there is a bunch of design choices need to consider. The first one is: where should the recurrent information be extracted from. It could be extracted from the cell c_t or cell output m_t, or from the penultimate layer r_t or p_t, or from the output y_t. Another question is: which computation block will receive the recurrent information? The information could simply be augmented to the input variable x_t, but could also be passed on to the input gate i_t, the output gate o_t, the forget gate f_t or the non-linear function $g(\cdot)$. Finally, the computing block that the information is extracted from is not necessarily the same for different tasks, nor is the block that receives the information. However in this study, we consider only the symmetric structure for simplicity.

With all the above alternative options, the multi-task recurrent model is rather flexible. The cognition evidence doesn't suggest any particular structure, although some general principles exist (e.g., the information exchange probably takes place during some intermediate steps rather than from the very beginning). For this reason, we need to experiment with a few alternate configurations of the network structure.

An example structure is shown in Fig. 4, where the recurrent information is extracted from both the recurrent projection r_t and the non-recurrent projection p_t, and the information is used as additional input to the non-linear function $g(\cdot)$. We use the superscript a and s to denote the ASR and SRE tasks respectively. The

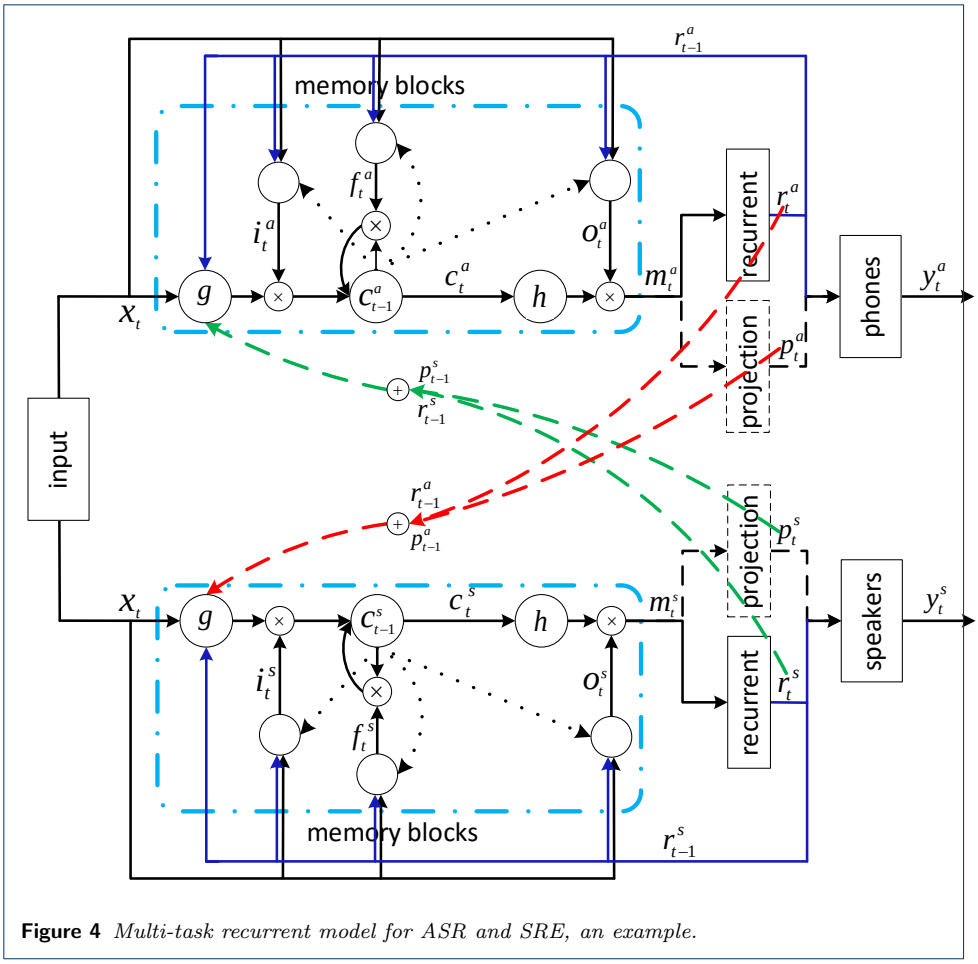


Figure 4 *Multi-task recurrent model for ASR and SRE, an example.*

computation for ASR can be expressed as follows (SRE recurrent information is underlined):

$$
\begin{aligned}
i_t^a &= \sigma(W_{ix}^a x_t + W_{ir}^a r_{t-1}^a + W_{ic}^a c_{t-1}^a + b_i^a) \\
f_t^a &= \sigma(W_{fx}^a x_t + W_{fr}^a r_{t-1}^a + W_{fc}^a c_{t-1}^a + b_f^a) \\
g_t^a &= g(W_{cx}^a x_t^a + W_{cr}^a r_{t-1}^a + b_c^a + \underline{W_{cr}^{as} r_{t-1}^s + W_{cp}^{as} p_{t-1}^s}) \\
c_t^a &= f_t^a \odot c_{t-1}^a + i_t^a \odot g_t^a \\
o_t^a &= \sigma(W_{ox}^a x_t^a + W_{or}^a r_{t-1}^a + W_{oc}^a c_t^a + b_o^a) \\
m_t^a &= o_t^a \odot h(c_t^a) \\
r_t^a &= W_{rm}^a m_t^a \\
p_t^a &= W_{pm}^a m_t^a \\
y_t^a &= W_{yr}^a r_t^a + W_{yp}^a p_t^a + b_y^a
\end{aligned}
$$

and the computation for SRE is as follows (ASR recurrent information is underlined):

$$
\begin{aligned}
i_t^s &= \sigma(W_{ix}^s x_t + W_{ir}^s r_{t-1}^s + W_{ic}^s c_{t-1}^s + b_i^s) \\
f_t^s &= \sigma(W_{fx}^s x_t + W_{fr}^s r_{t-1}^s + W_{fc}^s c_{t-1}^s + b_f^s) \\
g_t^s &= g(W_{cx}^s x_t^s + W_{cr}^s r_{t-1}^s + b_c^s + \underline{W_{cr}^{sa} r_{t-1}^a + W_{cp}^{sa} p_{t-1}^a}) \\
c_t^s &= f_t^s \odot c_{t-1}^s + i_t^s \odot g_t^s \\
o_t^s &= \sigma(W_{ox}^s x_t^s + W_{or}^s r_{t-1}^s + W_{oc}^s c_t^s + b_o^s) \\
m_t^s &= o_t^s \odot h(c_t^s) \\
r_t^s &= W_{rm}^s m_t^s \\
p_t^s &= W_{pm}^s m_t^s \\
y_t^s &= W_{yr}^s r_t^s + W_{yp}^s p_t^s + b_y^s
\end{aligned}
$$

4.3 Full and partial collaborative training

The multi-task recurrent model described above can be trained with complete or incomplete data. Complete data are labelled by both speech and speaker targets for each training sample; incomplete data, in contrast, are labelled by only one particular target, either speech or speaker. We call the training with complete data 'full collaborative training', and the training with incomplete data 'partial collaborative training'. In the previous study [24], we have shown that full collaborative training can improve performance on both ASR and SRE, however partial collaborative training has not been fully investigated. Since most existing large-scale databases are designed for a particular task and so are labelled incompletely, empirical evidence for partial collaborative training is very important.

Intuitively, partial collaborative training is analogous to teaching a child different knowledge types one at a time in a random and alternating manner, with the hope that each knowledge type contributes to the better understanding of other types. In practice, there are a number of issues that need to be addressed, e.g., how to balance the gradient contribution from each task, how the tasks impact each other through the recurrent connections, etc. We will discuss these issues in the following section.

5 Experiments

In this section, we first present the experiments with full collaborative training, and then report experiments with partial collaborative training, where the ASR and SRE components were trained with the Switchboard (SWB) and Fisher database respectively. All the experiments were conducted with the Kaldi toolkit [45].

5.1 Full collaborative training

In the first experiment, we study full collaborative training where the training data are labeled with both speech content (phones) and speaker identifies. The costs on both speaker and phone targets are added together, and the gradient on the accumulated cost is computed and back-propagated to update the entire network. WSJ database has been used in this experiment. As opposed to the preliminary

results presented in our earlier work [24], the results presented here use the WSJ corpus down-sampled to 8k Hz. This was done for uniformity in sampling rate across all the databases used in this study, viz., WSJ, Switchboard, Fisher and Eval2000.

5.1.1 Data

- Training set: This set comprises all the data in train_si284 consisting of 282 speakers and 37,318 utterances, with about 50-155 utterances per speaker. This set was used to train an ASR baseline based on LSTM, and two SRE baselines based on LSTM and i-vectors respectively. The same data were also used to train the proposed multi-task recurrent model.
- Test set: This set includes three datasets (devl92, eval92 and eval93). It consists of 27 speakers and 1,049 utterances in total. It was used to evaluate the performance of both ASR and SRE. For SRE, the evaluation consists of 21,350 target trials and 528,326 non-target trials, constructed from the speakers in the test set.

5.1.2 ASR baseline

The ASR system was built largely following the Kaldi WSJ s5 nnet3 recipe, except that we have used a single LSTM layer for simplicity. The number of cells in the hidden layer was set to 1,024, and the dimensions of the recurrent and non-recurrent projections were set to 256. The natural stochastic gradient descent (NSGD) algorithm [46] was employed to train the model. The input feature was 40-dimensional Fbanks, with a symmetric 2-frame window to splice neighboring frames. The output layer consisted of 3,377 units, equal to the total number of Gaussians in the conventional GMM system used to bootstrap the LSTM model. The language model is the WSJ official full trigram model ('tgpr') comprising 19,982 words. The baseline word error rate (WER) is 10.30%.

5.1.3 SRE baseline

We built two SRE baseline systems: one is an i-vector system and the other is an 'r-vector' system that is based on the recurrent LSTM model. For the i-vector system, the acoustic feature was 60-dimensional MFCCs. A UBM with 2,048 Gaussian components was used and the dimension of the i-vectors was set to 200. For the r-vector system, the architecture is similar to the one used by the LSTM-based ASR baseline, except that the number of cells is 512, and the dimensions of the recurrent and non-recurrent projections have been set to 128. These values were found empirically. The input of the r-vector system is the same as ASR system (Fbanks), and the output corresponded to the 282 speakers in the training set. Similar to the work in [3, 4], the speaker vector ('r-vector') was derived from the output of the recurrent and nonrecurrent projections, by averaging the output of all the frames. The dimension was 256. Voice activity detection (VAD) was employed before feature extraction in the single-task baseline.

The baseline performance is reported in Table 1 in terms of equal error rate (EER). It can be observed that the i-vector system generally outperforms the r-vector system. Particularly, the discriminative models (LDA and PLDA) offer significant improvement for the i-vector system compared to the r-vector system. This

observation is consistent with the results reported in [4], and can be attributed to the fact that the r-vector model has already been learned 'discriminatively' with the LSTM structure. For this reason, in the following experiments we only consider the simple cosine scoring for r-vector systems.

Table 1 *WSJ baseline*

	EER%		
System	Cosine	LDA	PLDA
i-vector (200)	3.13	1.67	1.06
r-vector (256)	2.71	1.77	5.99

5.1.4 Multi-task collaborative training

Due to the flexibility of the multi-task recurrent LSTM structure, it is not possible to experiment with all the configurations. We chose some typical settings and report the results in Table 2, where the first row of the numbers report the baseline system. Note that the last configuration, where the recurrent information is fed to all the gates and the non-linear activation $g(\cdot)$, is equivalent to augmenting the auxiliary information to the input x. From our previous work [24], we observe that the recurrent projection presents sufficient feedback information, so we report systems with feedback from the recurrent projection only.

Table 2 *WSJ full collaborative training*

Feedback Input				ASR WER%	SRE EER%
i	f	o	g		
				10.30	2.71
√				9.68	0.67
	√			9.88	0.92
		√		9.82	0.96
			√	**9.65**	0.89
√	√	√		9.73	0.62
√	√	√	√	9.86	**0.57**

The results reported in Table 2 display trends consistent with the 16KHz WSJ joint system reported in [24]. We first observe that the multi-task recurrent model consistently improves performance on both ASR and SRE, no matter where the recurrent information is extracted from and where it is applied. Interestingly, on the SRE task, the joint system matches and even outperforms the i-vector/PLDA system with careful selection of the configuration. To the best of our knowledge, this is the first work where two collaborative tasks are learned jointly in a unified framework and boost each other.

For the recurrent information 'receiver', i.e., the computing blocks that the recurrent information is applied to, it seems that for ASR the input gate and the activation function are the most effective, while enhancing the output gate doesn't appear so effective. For SRE, all the configurations seem good. These observations are just based on a relatively small database and it would be interesting to see if these findings generalize to larger data sets. Note that the performance improvement obtained with the collaborative training cannot be attributed to the enlargement

of the network. For thoroughness, an experiment doubling the hidden units lead to just a marginal performance gain on ASR, and the performance was even worse on SRE, probably due to over-fitting problem.

5.1.5 Comparison with structure sharing

We compare collaborative learning with the conventional structure sharing approach (Fig. 2(a)). To make the comparison, the simple single-layer LSTM structure of the baseline ASR and SRE systems is enhanced by adding four full-connection (FC) layers to learn deep features. Three systems are constructed: (1) ASR and SRE single task systems; (2) ASR and SRE joint learning system with the four FC layers shared; (3) ASR and SRE collaborative learning system with the four FC layers shared, and the LSTM layer collaboratively trained. Table 3 presents the results, where the best configurations from Table 2 are chosen for the collaborative learning system. It can be seen that the feature sharing approach does not provide clear performance gains over single task systems, while the collaborative learning provides comparable performance improvement as in Table 2. This confirms our conjecture that ASR and SRE are information-competitive tasks, and therefore hardly benefit from structure sharing. Collaborative learning is a more appropriate joint training approach for these tasks.

Table 3 *Comparison with structure sharing*

System	ASR WER%	SRE EER%
Single task	9.41	0.51
Structure sharing	9.40	0.64
Collaborative learning-g	9.06	0.52
Collaborative learning-ifog	9.29	0.47

5.1.6 Comparison with context-aware models

It has been demonstrated that involving phone posteriors can improve NN-based SRE [4], and involving speaker vectors improves NN-based ASR [5, 6]. These context-aware neural models are similar to the recurrent mulitask model, except that the two tasks are not collaboratively trained and conducted. Comparing these context-aware approaches with the collaborative learning approach will reveal the sole contribution of the collaboration mechanism.

To make the conclusion more concrete, we built a context-aware ASR (CA-ASR) system and a context-aware SRE (CA-SRE) system following the structure of the collaborative learning system, but removing the recurrent connections. The baseline ASR system is used to provide the context information for the CA-SRE system, and the baseline SRE system is used for the CA-ASR system. Referring to the best ASR and SRE configuration in Table 2, the speaker information is fed into the activation function g, and the phone information is fed into i, f, o, g.

The results are shown in Table 4. Firstly, it can be seen that the CA-ASR system outperforms the baseline, while the CA-SRE performs worse than the baseline. These results suggest that context-aware models may lead to performance gains, but this is not necessary and the result depends on how the context information

is involved. Secondly, it can be observed that the collaborative learning system outperforms the context-aware systems on both ASR and SRE, confirming that the collaboration mechanism indeed contributes significantly.

Table 4 *Comparison with Non-collaborative system*

	System	ASR WER%	SRE EER%
ASR	Baseline	10.30	-
ASR	CA-ASR	9.97	-
ASR	iv-ASR	9.92	-
SRE	Baseline (r-vector)	-	2.71
SRE	CA-SRE	-	3.06
SRE	i-vector / PLDA	-	1.06
SRE	DNN i-vector / PLDA	-	1.00
Collaborative learning-g		9.65	0.89
Collaborative learning-ifog		9.86	0.57

Finally, we built another two context-aware systems: a DNN i-vector SRE system [7, 8, 9, 10, 11], where phone posteriors (generated from the ASR baseline) are used to accumulate the Baum-Welch statistics; and an i-vector ASR system (iv-ASR) [5, 6] where segment i-vectors are augmented to the filter-bank input. The length of the segment has been empirically set to 400 frames. These two systems are not fully neural but indeed utilize auxiliary information, so can be compared with the collaborative learning. The results are also shown in Table 4. It can be observed that all these context-aware approaches provide reasonable performance gains (the DNN i-vector system outperforms the i-vector system, and the iv-ASR system outperforms the ASR baseline), but the collaborative learning performs the best. This demonstrates that the collaboration mechanism is an appropriate way to leverage the mutual information of collaborative tasks.

5.2 Partial collaborative training

In order to understand the implications of partial collaborative training where the data are labelled for only one of the tasks we conduct two sets of experiments; 1) We adapt the joint system in section 5.1 with partially labeled databases with just one of the ASR or SRE targets. 2) The two databases are then combined for a 'complete' partial collaborative training.

5.2.1 Data

- SWB: This database was mainly used to improve the ASR component of the joint system. It involves 313 hours of speech signals with word level transcriptions.
- Fisher: This database was mainly used to improve the SRE component of the joint system. It consists of $6,047$ utterances from $2,000$ speakers ($1,000$ females and $1,000$ males).
- Eval2000: This database was used to evaluate the performance of both ASR and SRE. It consists of 80 speakers and $4,458$ utterances. For SRE, the evaluation consists of $133,383$ target trials and $9,801,270$ non-target trials.

5.2.2 ASR-oriented partial collaborative training with SWB

In this experiment, the SWB database was used for partial collaborative training with focus on ASR component. Only the phonetic labels of the SWB corpus were considered in this experiment.

Three single-task systems were built as baselines: The first one is the 8kHz WSJ ASR baseline as presented earlier (WSJ); the second is an ASR baseline trained on SWB using the same recipe as the WSJ baseline (SWB); the third one is WSJ baseline adapted with the SWB corpus (WSJ+SWB), where the network of the WSJ baseline, except the last hidden layer, is reused, and the last hidden layer (with the speakers in SWB as the new targets) is re-initialized randomly.

For evaluation, the transcriptions of the SWB and Fisher databases were used to train a 3-gram LM, and Eval2000 was used as the test set. The WER results of these baseline systems are shown in Table 5. We observe similar results when comparing the SWB system and the WSJ+SWB system. The WSJ system is worse, which can be attributed to mismatch in the channel conditions of WSJ and Eval2000.

Table 5 *ASR single-task baselines for partial collaborative training*

	WER%
WSJ	58.6
SWB	24.0
WSJ+SWB	23.9

For partial collaborative training, we used SWB to conduct the ASR-oriented partial training based on the joint system trained with WSJ. For simplicity, only the input gate and the nonlinear function were used to receive the recurrent information. The results are shown in Table 6. It can be seen that the partial collaborative training adapts the joint system and improves the ASR performance significantly compared to the WSJ-initialized joint system, and it also outperforms all the three single-task ASR baselines.

The SRE results of the adapted system are also shown in Table 6. It is interesting to see that, although the training data do not involve any speaker labels, the SRE performance is improved. We conjecture that this is because the more accurate recurrent information offered by the improved ASR component. This is a nice property and supports well our argument that the recurrent model is suitable for modeling collaborative tasks, and improvement on one task may benefit other tasks.

Table 6 *ASR-oriented partial collaborative training with SWB*

Feedback Input				WSJ Initialized	SWB Trained	ASR WER%	SRE EER%
i	f	o	g				
√				√		58.9	21.32
			√	√		58.7	21.56
√				√	√	22.9	18.66
			√	√	√	22.9	18.10

5.2.3 SRE-oriented partial collaborative training with Fisher

In this experiment, the Fisher database was used to conduct the partial collaborative training, where only the speaker labels were used to adapt the joint system. Similar

to the ASR-oriented partial training experiment, three single-task r-vector SRE baselines were built: the first one is the WSJ r-vector baseline (WSJ), and the second was trained from scratch using the Fisher database (Fisher), following the same recipe used by the WSJ baseline, and the third one is the WSJ baseline adapted with the Fisher database (WSJ+Fisher). The test set is Eval2000. The EER results of the baseline systems are shown in Table 7. We see that the Fisher baseline performs the best, and the WSJ initial system does not help (actually it leads to worse performance). Again, this can be attributed to the mismatch in channel conditions.

Table 7 *SRE single-task baselines for partial collaborative training*

	EER%	
	i-vector / PLDA	r-vector / Cosine
WSJ	20.49	24.68
Fisher	16.97	15.95
WSJ+Fisher	15.77	16.82

The partial collaborative training also starts from the WSJ joint system, which is followed by the SRE-oriented partial collaborative training using the Fisher database. The results are shown in Table 8. It can be observed that the initial WSJ joint system performs worse than the single-task WSJ baseline on the SRE task. With the SRE-oriented partial collaborative training, the SRE performance is significantly improved, even better than those obtained with the single-task baselines. Unfortunately, the SRE-oriented partial training does not seem to help ASR; it in fact deteriorates the ASR component significantly. We conjecture that the worse performance is due to SRE-oriented 'overfitting'. Specifically, the model parameters may have been adapted over-aggressively to improve SRE, resulting in undesirable parameter changes in the ASR component, resulting in very poor performance.[1] Interestingly, the bad ASR still leads to a strong SRE performance, which means that inaccurate ASR information can still boost the SRE performance, otherwise the joint system can not beat the single-task baselines.

We get interesting insights when the results are compared to those in ASR-oriented training presented in Table 6. In the ASR-oriented training, both ASR and SRE components are improved, while in the SRE-oriented training, only the SRE component is improved. This suggests that the two tasks not only collaborative but also competitive in the collaborative learning. On one hand, they boost each other by sharing information, and on the other hand, they tried to optimize their individual objectives. A subtle trade-off needs to be considered to balance the effect of the two aspects, which becomes more clear in the complete partial collaborative training presented in the next experiment.

5.2.4 Partial collaborative training with both SWB and Fisher

In the final experiment, we combine the SWB and Fisher databases (with partial labels) for complete partial collaborative training. The speaker-labeled data and

[1]Note that ASR components are also updated even though the training data contain only speaker labels, due to gradient propagation through the recurrent connections.

Table 8 *SRE-oriented partial collaborative training with Fisher*

Feedback Input				WSJ Initialized	Fisher Trained	ASR WER%	SRE EER%
i	f	o	g				
√				√		58.9	21.32
			√	√		58.7	21.56
√				√	√	96.2	10.90
			√	√	√	96.1	11.63

phone-labeled data are mixed together and are input sequentially in mini-batches. Similar to the single-task-oriented partial collaborative training experiments, we can use the joint system trained with WSJ as the initial model and use the two databases to adapt the system. However, since phone targets and speaker targets are both available (although not with a single sample), we can train the joint system from scratch. Our experiments show that the two approaches lead to similar performance, so we just report the results with the system trained from scratch.

As mentioned in the previous experiment, ASR and SRE are both collaborative and competitive, so their relative contribution to the collaborative training should be balanced. In our experiments, the Fisher database is about 3 times as large as the SWB database. This means that the partial collaborative training might be biased towards the SRE training. To investigate the impact of the bias, we conducted four sets of experiments, where the amount of the SRE data extracted from Fisher varied from 0.5 to 3 times of the SWB data. The data selection was utterance-based, and the number of target speakers was constant.

The baseline systems are the single-task ASR and SRE systems trained with SWB and Fisher respectively, as shown in Table 5 and Table 7, respectively. Table 9 shows the results with the complete partial collaborative training. For a clear comparison, the baseline results are also presented. It can be observed that the trend of the performance is like a combination of the results in Table 6 and Table 8, taking into account the relative data volume for the two tasks. Specifically, more ASR-oriented training generally improves both ASR and SRE, and more SRE-oriented training improves SRE more significantly, but hurts ASR. Importantly, an appropriate data ratio (e.g., 1:0.5) leads to good performance for both tasks: it obtains the best ASR result, and a very competitive SRE result. Focusing on each single task, the complete partial collaborative training obtains the best performance on each of them (22.6 on ASR and 10.54 on SRE). Interestingly, the performance on SRE does not increase monotonically with more Fisher data. This can be attributed to the collaborative and competitive mechanism: too much speaker data leads to better SRE performance but hurts ASR, and the worse ASR performance may in turn impact SRE. In other words, a good ASR benefits SRE in the complete partial training. This is very different from the SRE-oriented partial training where SRE is the only goal of the training so that it can be improved even if the ASR performance is bad. We therefore conclude that the SRE performance can be improved in two ways: by complete partial training (as in Table 9) to seek for better auxiliary information, or by SRE-oriented partial training (as in Table 8) to seek for SRE-oriented optimization. These two approaches have different rationalities and their optimal configurations are clearly distinct, and which model is optimal is undetermined.

Table 9 *Complete partial collaborative training with both SWB and Fisher*

Feedback Input				SWB & Fisher Ratio	ASR WER%	SRE EER%
i	f	o	g			
-				-	24.0 (SWB)	15.95 (Fisher)
√				1 : 0.5	23.2	11.44
			√	1 : 0.5	**22.6**	10.91
√				1 : 1	24.2	11.09
			√	1 : 1	23.2	10.85
√				1 : 2	26.0	**10.54**
			√	1 : 2	24.5	11.07
√				1 : 3	28.8	11.13
			√	1 : 3	26.5	12.05

5.3 Discussion

The experiments presented in this section clearly demonstrate the capacity of collaborative learning with the multi-task recurrent model. However, to make the model most effective, there are several important issues that need to be addressed. First of all, as the information propagation paths are rather flexible, and the design of such paths largely depends on the structure of each individual component, it is hard to predict which configuration is optimal. In our experiments, the input gate and the nonlinear activation function seem the best to receive the recurrent information, but for other tasks (e.g., language recognition and speaker recognition) and other structures (e.g., vanilla RNN), the propagation paths need to be carefully chosen and the optimal design can be only determined by experiments. This can be a time consuming task. A better approach involving either parameter shrinkage or prior knowledge from cognitive study is desirable.

Another issue that needs to be addressed is the structure and configuration for individual components in joint systems. Due to the recurrent structure, optimal models/configurations may be different from the ones used in single-task systems. For example, in our experiments voice activity detection (VAD) works well in the single-task SRE systems, but in the joint system, VAD leads to worse performance. This is possibly attributed to the signal discontinuity caused by VAD, which results in worse ASR and hence worse SRE.

Finally, different tasks may behave very differently in collaborative training. For instance, in our experiments SRE learning seems more aggressive than ASR learning: a relatively small amount of speaker-labeled data may lead to a clearly SRE-biased joint model. This 'asymmetry' among tasks is closely related to the collaboration and competition mechanism, and can be attributed to a multitude of factors for the two tasks, e.g., the cost functions, the separability among targets, and the strength of the gradient back propagated through the recurrent connections. In our experiments, control of the relative contribution of each task in the collaborative training was simply based on empirical evidence, but more theoretical ways would be desirable.

Acknowledgement

This work was also partly supported by Sinovoice and Huilan Ltd. Thanks to Gary Cook for valuable suggestions.

6 Conclusions

We have proposed a novel collaborative learning approach based on multi-task recurrent neural model, and applied this approach for joint learning of speech and speaker recognition tasks. A thorough empirical investigation was conducted and the results demonstrated that the presented approach can learn speech and speaker models in a joint way and can improve the performance on both tasks. In particular, we have demonstrated the feasibility of collaborative training with partially-labeled data, which emphasizes the practical value of this approach. Future work involves further investigation on the collaborative structure and the task asymmetry. Applying the method to other collaborative tasks will also be explored.

Author details

[1]Center for Speech and Language Technology, Research Institute of Information Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China. [2]Center for Speech and Language Technologies, Division of Technical Innovation and Development, Tsinghua National Laboratory for Information Science and Technology, ROOM 1-303, BLDG FIT, 100084 Beijing, China. [3]Department of Computer Science and Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China. [4]Nuance, UK..

References

1. G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.
2. G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.
3. E. Variani, X. Lei, E. McDermott, I. Lopez Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.
4. L. Li, Y. Lin, Z. Zhang, and D. Wang, "Improved deep speaker feature learning for text-dependent speaker recognition," in *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2015, pp. 426–429.
5. A. W. Senior and I. Lopez-Moreno, "Improving DNN speaker independence with i-vector inputs," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 225–229.
6. G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *ASRU*, 2013, pp. 55–59.
7. M. K. Omar and J. W. Pelecanos, "Training universal background models for speaker recognition," in *Proceedings of Odyssey*, 2010, pp. 52–57.
8. Y. Lei, L. Ferrer, M. McLaren *et al.*, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1695–1699.
9. P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting Baum-Welch statistics for speaker recognition," in *Proceedings of Odyssey*, 2014, pp. 293–298.
10. S. O. Sadjadi, S. Ganapathy, and J. W. Pelecanos, "The ibm 2016 speaker recognition system," in *Proceedings of Odyssey*, 2016.
11. F. Richardson, D. Reynolds, and N. Dehak, "Deep neural network approaches to speaker and language recognition," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1671–1675, 2015.
12. M. F. BenZeghiba and H. Bourlard, "On the combination of speech and speaker recognition," in *Proceedings of European Conference On Speech, Communication and Technology (EUROSPEECH)*, no. EPFL-CONF-82941, 2003, pp. 1361–1364.
13. A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of International Conference on Machine Learning (ICML)*, 2014, pp. 1764–1772.
14. H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 2014, pp. 338–342.
15. G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5115–5119.
16. R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.
17. D. Wang and T. F. Zheng, "Transfer learning for speech and language processing," in *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2015, pp. 1225–1237.
18. J.-T. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 7304–7308.

19. G. Heigold, V. Vanhoucke, A. Senior, P. Nguyen, M. Ranzato, M. Devin, and J. Dean, "Multilingual acoustic models using distributed deep neural networks," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 8619–8623.
20. A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep neural networks," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 7319–7323.
21. S. Thomas, M. L. Seltzer, K. Church, and H. Hermansky, "Deep neural network features and semi-supervised training for low resource speech recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 6704–6708.
22. K. M. Knill, M. J. Gales, A. Ragni, and S. P. Rath, "Language independent and unsupervised acoustic models for speech recognition and keyword spotting," in *Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 2014, pp. 16–20.
23. N. Chen, Y. Qian, and K. Yu, "Multi-task learning for text-dependent speaker verification," in *Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 2015, pp. 185–189.
24. Z. Tang, L. Li, and D. Wang, "Multi-task recurrent model for speech and speaker recognition," *arXiv preprint arXiv:1603.09643*, 2016.
25. X. Li and X. Wu, "Modeling speaker variability using long short-term memory networks for speech recognition," in *Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 2015, pp. 1086–1090.
26. A. S. Walker-Andrews, L. E. Bahrick, S. S. Raglioni, and I. Diaz, "Infants' bimodal perception of gender," *Ecological Psychology*, vol. 3, no. 2, pp. 55–75, 1991.
27. D. M. Houston and P. W. Jusczyk, "The role of talker-specific information in word segmentation by infants," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 26, no. 5, pp. 1570–1582, 2000.
28. ——, "Infants' long-term memory for the sound patterns of words and voices," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 29, no. 6, pp. 1143–1154, 2003.
29. E. K. Johnson, E. Westrek, T. Nazzi, and A. Cutler, "Infant ability to tell voices apart rests on language experience," *Developmental Science*, vol. 14, no. 5, pp. 1002–1011, 2011.
30. J. F. Werker and S. Curtin, "Primir: A developmental framework of infant speech processing," *Language Learning and Development*, vol. 1, no. 2, pp. 197–234, 2005.
31. C. D. Creelman, "Case of the unknown talker," *Journal of the Acoustical Society of America*, vol. 29, no. 5, p. 655, 1957.
32. Q. Summerfield and M. Haggard, "Vocal tract normalization as demonstrated by reaction times," *Report of Speech Research in progress*, vol. 2, pp. 12–23, 1973.
33. R. R. Verbrugge, W. Strange, D. P. Shankweiler, and T. R. Edman, "What information enables a listener to map a talker's vowel space?" *Journal of the Acoustical Society of America*, vol. 60, no. 1, pp. 198–212, 1976.
34. J. W. Mullennix, D. B. Pisoni, and C. S. Martin, "Some effects of talker variability on spoken word recognition," *Journal of the Acoustical Society of America*, vol. 85, no. 1, pp. 365–378, 1989.
35. H. C. Nusbaum and T. M. Morin, "Paying attention to differences among talkers," *Speech Perception, Speech Production, and Linguistic Structure*, pp. 113–134, 1992.
36. K. Johnson, "Speaker normalization in speech perception," *The Handbook of Speech Perception*, pp. 363–389, 2008.
37. I. Eklund and H. Traunmüller, "Comparative study of male and female whispered and phonated versions of the long vowels of swedish," *Phonetica*, vol. 54, no. 1, pp. 1–21, 1997.
38. L. C. Nygaard, "Perceptual integration of linguistic and nonlinguistic properties of speech," *The Handbook of Speech Perception*, pp. 390–413, 2008.
39. P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1435–1447, 2007.
40. X. Zeng, S. Yin, and D. Wang, "Learning speech rate in speech recognition," in *Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 2015, pp. 528–532.
41. Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
42. G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
43. Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle *et al.*, "Greedy layer-wise training of deep networks," *Advances in Neural Information Processing Systems*, vol. 19, pp. 153–160, 2007.
44. Y. Bengio *et al.*, "Deep learning of representations for unsupervised and transfer learning," *Journal of Machine Learning Research: Workshop and Conference Proceedings*, vol. 27, pp. 17–36, 2012.
45. D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, and P. Schwarz, "The kaldi speech recognition toolkit," in *Proceedings of IEEE 2011 workshop on Automatic Speech Recognition and Understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.
46. D. Povey, X. Zhang, and S. Khudanpur, "Parallel training of dnns with natural gradient and parameter averaging," *arXiv preprint arXiv:1410.7455*, 2014.