	CNN		DailyMail	
	Dev	Test	Dev	Test
Hermann et al. (2015)	61.6	63.0	70.5	69.0
Hill et al. (2015)	63.4	6.8	-	-
Kadlec et al. (2016)	68.6	69.5	75.0	73.9
Kobayashi et al. (2016)	71.3	72.9	-	-
Sordoni et al. (2016)	72.6	73.3	-	-
Trischler et al. (2016)	73.4	74.0	-	-
Chen et al. (2016)	73.8	73.6	77.6	76.6
Cui et al. (2016)	73.1	74.4	-	-
Shen et al. (2016)	72.9	74.7	77.6	76.6
BIDAF	76.31	76.94	80.33	79.63
CS-BIDAF	76.43	77.10	80.37	79.71
IC-BIDAF	76.41	77.21	80.49	79.83
MA-BIDAF	76.49	77.09	80.42	79.74
DC-BIDAF	76.35	77.15	80.38	79.67
AC-BIDAF	76.62	77.23	80.65	79.88
Dhingra et al. (2016)	77.9	77.9	81.5	80.9
Dhingra et al. (2017)	79.2	78.6	–	–

Table 4. Accuracy (%) on the two QA datasets

each containing a training, development and test set with 300k/4k/3k and 879k/65k/53k examples respectively. Each example consists of a passage, a question and an answer. The question is a cloze-style task where an entity is replaced by a placeholder and the goal is to infer this missing entity (answer) from all the possible entities appearing in the passage. The LSTM architecture and experimental settings follow the Bidirectional Attention Flow (BIDAF) (Seo et al., 2017) model, which consists of the following layers: character embedding, word embedding, contextual embedding, attention flow, modeling and output. LSTM is applied in the contextual embedding and modeling layer. Character embedding is based on one-dimensional convolutional neural network, where the number of filters is set to 100 and the width of receptive field is set to 5. In LSTM, the size of hidden state is set to 100. Optimization is based on AdaDelta (Zeiler, 2012), where the minibatch size and initial learning rate are set to 48 and 0.5. The model is trained for 8 epochs. Dropout (Srivastava et al., 2014) with probability 0.2 is applied. We compare with four diversity promoting regularizers: CS, IC, MA and DC.

Table 4 shows state of the art accuracy on the two datasets. As can be seen, after applying AC to BIDAF, the accuracy is improved from 76.94% to 77.23% on the CNN test set and from 79.63% to 79.88% on the DailyMail test set. Among the diversity-promoting regularizers, AC achieves the highest accuracy.

5.3. Sensitivity to Parameter τ

In the theoretical analysis presented in Section 4, we have shown that the parameter τ which controls the level of near-orthogonality (or diversity) incurs a tradeoff between estimation error and approximation error. In this section, we provide an empirical verification, using FNN on TIMIT as a study case. Figure 1 shows how the phone error rates vary on the TIMIT core test set. As can be seen, the lowest test error is achieved under a moderate τ $(= 0.75)$. Either a smaller or a larger τ degrades the performance. This empirical observation is aligned with the theoretical analysis that the best generalization performance is achieved under a properly chosen τ. When τ is close to 0, the hidden units are close to orthogonality, which yields much poorer performance. This confirms that the strict-orthogonality constraint proposed by (Le et al., 2010) is too restrictive and is less favorable than a "soft" regularization approach.

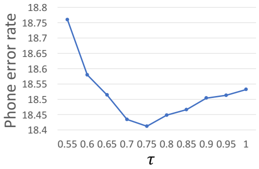

Figure 1. Phone error rate on TIMIT, under varying τ

	TIMIT	CIFAR-10	CNN
No regularization	1.1	6.3	69.4
CS	1.2	6.8	74.8
IC	1.2	6.7	76.1
MA	1.3	7.0	78.6
DC	1.5	7.6	82.9
OP	–	6.8	–
AC	1.3	7.1	79.7

Table 5. Average runtime (hours)

5.4. Computational Time

We compare the computational time of neural networks under different regularizers. Table 5 shows the total runtime time of FNNs on TIMIT and CNNs on CIFAR-10 with a single GTX TITAN X GPU, and the runtime of LSTM networks on the CNN dataset with 2 TITAN X GPUs. Compared with no regularization, AC incurs a 18.2% extra time on TIMIT, 12.7% on CIFAR-10 and 14.8% on CNN. The runtime of AC is comparable to that under other diversity-promoting regularizers.

6. Conclusions

In this paper, we propose Angled-Constrained Latent Space Models (AC-LSMs) that aim at promoting diversity among components in LSMs for the sake of alleviating overfitting. Compared with previous diversity-promoting methods, AC has two benefits. First, it is theoretically analyzable: the generalization error analysis shows that larger diversity leads to smaller estimation error and larger approximation error. Second, it is empirically effective, as validated in various experiments.