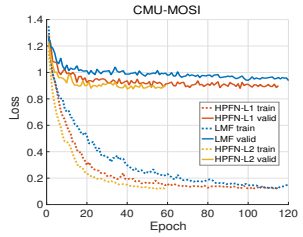
We thank all Reviewers for their constructive comments and insightful suggestions.



To Reviewer 1:

(1) Thank you for the very positive comments on our paper and kind suggestions about TRECVID dataset, we believe more impressive results can be provided in final version.

To Reviewer 2 & Reviewer 3:

(1) Parameter complexity: for PTP, due to the symmetry of feature tensor as well as the symmetric weight tensor, the number of parameters is independent of order P and linearly scales with the concatenated mixed features. For L-layer HPFN, number of parameters is linearly related to the number of 'windows' $\sum_{l=1}^{L} N_l$. In practice, N_l is usually small and decreasing along layers, e.g. $N_1 > N_2 > ... > N_L$. In our tests, complete/partial sharing strategy makes N_l even smaller. In principle, the parameter of HPFN is larger than or comparable to LMF (as HPFN is more powerful with temporal modelling), but significantly less than TFN. Please refer to the Table in this page.

The tradeoff is, if we employ more layers (or with more intermediate nodes in each layer) we get greater expressive capability. In practice, we need to choose optimal one so as not to overfit.

(2) Time complexity: PTP is comparable or similar to LMF; HPFN is less than $\sum_{l=1}^{L} N_l$ times of LMF, depending on specific architecture design choices such as number of layers, number of windows, window size and etc.

(3) About training details, such as hyper-parameters settings and number of parameters, will be added in final version.

To Reviewer 2:

(1) Thanks for detailed comments on clarity of paper (such as maths layout, concise descriptions of HPFN, model architecture table), we will include them in the final version.

(2) Regarding training curves, we illustrate HPFN L1 & L2 and LMF in Figure in this page showing that HPFN is better than LMF. More comparisons on training curves will be added in the final version.

(3) It is a great suggestion to include significance testing, such as p-value test, in the paper.

To Reviewer 3:

(1) Regarding the concept of 'local interactions', it refers to interactions of the concatenated features, from a time window and a subset of modalities located within a 'local window' (analogous to the 'convolution filter' of CNN). Please see 'bounding rectangles' in Figure 3 in paper as an example. We will improve the clarity by rewriting this part.

(2) The originality of our work includes: (a) higher order interactions and (b) local temporal information fusion can be achieved by PTP. HPFN can be explained as a CNN-style hierarchical fusion framework where the convolution operation is replaced by the PTP operator, to identify cross-modal interactions through time sequence.

(3) Regarding more baselines, MMB1&2 [NAACL19'] are two wonderful baselines that learn embeddings of multimodal utterances, we will add them in the next version. ConAC is based on convolutional operator and suitable for the conventional image recognition task. It does not directly consider the deep fusion of multimodal data, hence might not perform well. But it is interesting to adapt/modify ConAC to multimodal setting in the future work.

(4) Regarding imperfect data, our current model does not take imperfect data into account, but it is very interesting direction to work on. A very recent T2FN [ACL19'] is an excellent approach for imperfect data. We will refer this work and believe that incorporating its novel idea of tensor regularizer can make our method more robust to noisy data. The sharing strategy among multiple 'fusion filters' at each layer might be another possible option.

(5) Regarding the depth, we test HPFN up to 4 layers (Table 2 in paper) and the optimal number of layers for IEMOCAP and MOSI are 2 and 3, using the listed architectures. The optimal depth also depends on how each layer is designed.

(6) Regarding the order P, automatically determining the optimal P is nontrivial and interesting to investigate. Now the best way is to use CV. In our empirical test, the preferred order normally ranges from 4 to 8, which are related to specific tasks and datasets.

(7) Regarding performance improvement, as shown in Table 1 in paper, the high-order pooling and the number of layers (meaning more parameters) achieved similar amount of performance improvement over SOTA LMF method. We agree with Reviewer that it would be nice by adding more qualitative analysis in the final version.

Model	TFN [non-temporal]	LMF [non-temporal]	PTP [temporal]	HPFN (L layers) [temporal]
Parameter Complexity	$\mathcal{O}(I_y \prod_{m=1}^{M} I_m)$	$\mathcal{O}(I_y r(\sum_{m=1}^{M} I_m))$	$\mathcal{O}(I_y r(\sum_{t=1}^{T} \sum_{m=1}^{S} I_{t,m}))$	$\mathcal{O}(I_y r(\sum_{l=1}^{L} N_l)(\sum_{t=1}^{T} \sum_{m=1}^{S} I_{t,m}))$

Table 1: I_y is output length. M is number of modalities. r is tensor rank. For PTP, $[T, S]$ is the 'local window size' with $S \leq M$. $I_{t,m}$ is the dimension of features from modality m at time t. For HPFN, N_l is the number of PTP 'windows' at layer $l \in [L]$.