

# Interpretable Deep Learning for Monitoring Combustion Instability<sup>\*</sup>

Tryambak Gangopadhyay<sup>\*</sup> Sin Yong Tan<sup>\*</sup>  
Anthony LoCurto<sup>\*</sup> James B. Michael<sup>\*</sup> Soumik Sarkar<sup>\*</sup>

<sup>\*</sup> *Department of Mechanical Engineering, Iowa State University, Ames,  
IA 50011, USA  
(e-mail: tryambak, tsyong98, alocurto, jmichael, soumiks@iastate.edu).*

---

**Abstract:** Transitions from stable to unstable states occurring in dynamical systems can be sudden leading to catastrophic failure and huge revenue loss. For detecting these transitions during operation, it is of utmost importance to develop an accurate data-driven framework that is robust enough to classify stable and unstable scenarios. In this paper, we propose deep learning frameworks that show remarkable accuracy in the classification task of combustion instability on carefully designed diverse training and test sets. We train our model with data from a laboratory-scale combustion system showing stable and unstable states. The dataset is multimodal with correlated data of hi-speed video and acoustic signals. We develop a labeling mechanism for sequences by implementing Kullback–Leibler Divergence on the time-series data. We develop deep learning frameworks using 3D Convolutional Neural Network and Long Short Term Memory network for this classification task. To go beyond the accuracy and to gain insights into the predictions, we incorporate attention mechanism across the time-steps. This aids in understanding the time-periods which contribute significantly to the prediction outcome. We validate the insights from a domain knowledge perspective. By exploring inside the accurate black-box models, this framework can be used for the development of better detection frameworks in different dynamical systems.

*Keywords:* Deep Learning, Attention, LSTM, 3D CNN, Detection

---

## 1. INTRODUCTION

Developing highly accurate data-driven frameworks is important for the performance monitoring of dynamical systems leading to better control. The confidence in implementing these frameworks can be enhanced further if an accurate model can provide additional insights apart from generating the prediction. For temporal data-based systems, an explainable model highlighting the significant time-periods of the input sequence can give a better understanding when verified from the domain knowledge perspective. In combustion dependent power generating systems like land-based gas turbine engines, jet engines for aviation and rocket engines, combustion is generally operated in low-fuel states to achieve fuel economy and reduce nitrogen oxides emissions. But during such low-fuel operations, these systems become more susceptible to instability. Developing accurate and insightful frameworks is crucial for the detection of instability in combustion systems.

The phenomenon of combustion instability can be more frequent when the burning of fuel takes place in a confined environment. During such scenarios, flow perturbations

can result in fluctuating heat release rates leading to the generation of sound waves. These sound waves could vary the heat release rate when reflected from the walls. When the heat release rate fluctuations are in phase with the fluctuating acoustic pressure, a positive feedback loop can be established resulting in large amplitude oscillations (Rayleigh (1878); Dowling (1997)). During such intense growth of pressure fluctuations, engines can develop large levels of vibration causing catastrophic failure or drastically reducing the engine life (Fisher and Rahman (2009)). This can result in huge revenue loss and other serious consequences. Accurate detection of the stable and unstable states can lead to effective control and performance monitoring of a combustion system.

Previous works of studying combustion instability include using computational fluid dynamics (CFD) models, physics-based modeling (Palies et al. (2011)) among others. These models often have restrictions like simplifying assumptions, inherent complexities, computational restrictions, etc. Some data-driven methods such as proper orthogonal decomposition (POD) (Berkooz et al. (1993); LoCurto et al. (2018)), dynamic mode decomposition (DMD) (Schmid (2010); Ghosal et al. (2016)) are common, but they demonstrate high parametric sensitivity for detection of instability precursors. There are also other applications of data-driven techniques specifically for combustion systems (Nair and Sujith (2014); Sen et al. (2018, 2016)). However, implementing methods based on only

---

<sup>\*</sup> This work has been supported in part by the U.S. Air Force Office of Scientific Research under the YIP grant FA9550-17-1-0220. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agency.

acoustic data can sometimes be erroneous due to the hindering effects of combustion noise. Detection frameworks based on images can be more reliable, which we explore in this paper.

Deep learning (DL) models are becoming increasingly popular with widespread applications as these models can show highly accurate performance with no hand-designed features as input. DL models are superior to other state-of-the-art machine learning models (Hinton and Salakhutdinov (2006)). These DL models have initiated a rapid development in the field of computer vision. Convolutional Neural Network has become a popular architecture in computer vision and has demonstrated state-of-the-art results in learning meaningful features from images (Krizhevsky et al. (2012)). Long Short Term Memory (LSTM) can learn temporal dependencies effectively over long sequences (Hochreiter and Schmidhuber (1997)). The application of DL in the field of combustion instability has started recently and has shown promising results. The works include a neural-symbolic framework (Sarkar et al. (2015)) and frameworks based on convolutional selective autoencoder (Akintayo et al. (2016)), 2D CNN (Gangopadhyay et al. (2018)), 3D CNN (Ghosal et al. (2017)) and 2D CNN LSTMs (Gangopadhyay et al. (2020)). Using DL, the highest test accuracy scores reported earlier have been in the range of 83% - 85%. For the DL model which can provide insights (Gangopadhyay (2019)), there has been no validation of the insights from a domain knowledge perspective. Also, no previous work has explicitly shown robustness by achieving high accuracy in conditions showing different dynamics than that of the training set. In this paper, we aim to overcome these shortcomings of the previous works.

The contributions of this paper are:

- We propose an interpretable deep learning framework demonstrating remarkable accuracy and robustness to detect combustion instability in a test protocol having different dynamics to that of the training set.
- The insights learned using the attention mechanism of the framework are in accord with the domain knowledge.

## 2. PROBLEM FORMULATION AND DATASET

### 2.1 Data Collection and Preprocessing

For data collection, we use an experimental laboratory-scale setup (a vertically placed Rijke tube having two open ends) similar to that used by Gangopadhyay et al. (2018). We use two different lengths of the tube (2 ft, 4 ft), each with a diameter of 3 inches. The hi-speed flame video (at 5000 frames/sec) and acoustic signal are both recorded simultaneously. The acoustic length is varied by changing the position of the burner tip alternating the  $x/L$  ratio from 0.125 to 0.5, where  $x$  is the distance measuring from the bottom end of the tube to the bluff body. For both tubes (2ft, 4ft), data is collected at five different positions with  $x = 0$  in., 3 in., 6 in., 9 in., and 12 in.

At each position, we collect data twice independently with a time gap. The one-time record length of data is 12 secs (60,000 images). To ensure that the flame has

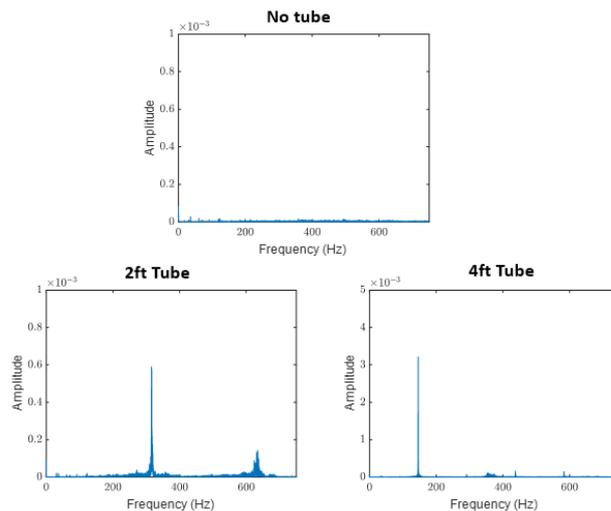


Fig. 1. Figure showing sample FFT plots for “No Tube”, “2 ft Tube” and “4 ft Tube” cases.

reached a steady-state, we wait for 10 seconds before beginning data collection at each position. We perform another experiment without the Rijke tube, which we label as the “no tube” case (baseline dataset). We have a total of 10 sets of data (5 different positions for each tube) and each set has two recordings (run *a*, run *b*). For easier experimental setup identification, we will address different conditions using the following convention: for example, the run *a* experiment using 4 ft tube, and with the position  $x = 0$  in., is denoted as 4ft.0\_run.a. After extracting the region of interest, each flame chemiluminescence image is resized to a resolution of 64 x 64. The resizing is done due to the restriction imposed by the GPU memory limits for training our 3D deep learning frameworks. The degree of downsampling the videos is a hyper-parameter in our modeling process which we will discuss more in the following sections. For preprocessing the time series, we implement a moving window approach (Sen et al. (2018)) with each time window of length 0.1 secs. Fast Fourier transform (FFT) is computed for each window, based on which we propose a labeling procedure.

### 2.2 Labeling Procedure

For our supervised learning model, we compute labels for the flame video snippets (image sequences) by incorporating domain knowledge and proposing a Kullback–Leibler (K-L) Divergence based metric. K-L Divergence is a statistical metric that measures the similarity between two distributions. By using the FFT plot of the “no tube” case (Fig. 1) as the baseline, we compute the K-L Divergence for each of the windowed FFT plots. We choose a range (+/- 50% of the fundamental frequency) considering that the amplitudes around fundamental frequency in the FFT plot is a strong indicator of instability from the domain knowledge perspective. The frequency range for the 4ft tube is therefore 75 - 225 Hz with 150 Hz as the fundamental frequency (Fig. 1). The results of the K-L Divergence computed for the 4 ft tube are shown in Fig. 2. Each point refers to a particular time window (0.1s) which in turn corresponds to the correlated image sequence. We use the two most separated clusters visible in the plot, labeling the 4\_ft.0\_run.a, 4\_ft.3\_run.a, 4\_ft.6\_run.a as the stable cases

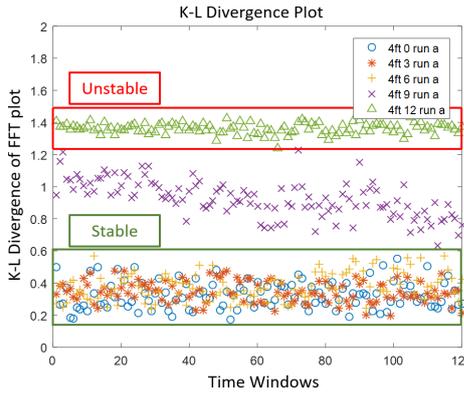


Fig. 2. The figure shows the K-L Divergence of FFT plot for each time window relative to the baseline FFT plot.

(green box) and 4\_ft\_12\_run.a as the unstable case (red box).

### 3. METHODOLOGY

#### 3.1 CNN, LSTM, Temporal Attention

Convolutional Neural Network (CNN) is a neural network architecture designed to capture the spatial correlation across image data. A CNN scans over an image with a receptive field (shared parameters) to first learn local features and then aggregate the local features. As the weights are shared, CNNs have less number of trainable parameters compared to fully connected layers. This leads to lower chances of overfitting and comparatively needs less computational power. By extracting meaningful features from images, CNNs have shown remarkable results in different image recognition tasks outperforming all other techniques (Krizhevsky et al. (2012); LeCun et al. (1998)).

Recurrent Neural Networks (RNNs) are capable of explicitly capturing temporal dependencies in sequences outperforming static networks (Bengio (1991)). To overcome the problem of vanishing gradients of RNNs for long sequences (Bengio et al. (1994); Gers et al. (1999)), Hochreiter and Schmidhuber (1997) proposed an effective RNN architecture Long Short Term Memory (LSTM). LSTM has input, output, and forget gates which prevent the perturbation of memory contents with irrelevant information. These gates regulate the addition of any information to the cell state of an LSTM block. Apart from input  $x^{<t>}$  at time-step  $t$ , an LSTM block takes in the cell state ( $C^{<t-1>}$ ) and hidden state ( $a^{<t-1>}$ ) of the previous time-step as inputs. The updated values of  $a^{<t>}$  and  $C^{<t>}$  act as inputs for the LSTM block of the next time-step.

For long sequences, the concept of soft temporal attention was introduced by Bahdanau et al. (2014) in the context of neural machine translation to overcome the bottleneck of the encoder-decoder model (Cho et al. (2014); Sutskever et al. (2014)), which compresses all information into a fixed-length vector. They fused the attention module between the encoder and decoder layers. However, in our proposed model, a decoder LSTM is not required as we are performing many-to-one classification. The temporal attention block takes in a sequence of hidden states as input and after aggregating the information, computes a

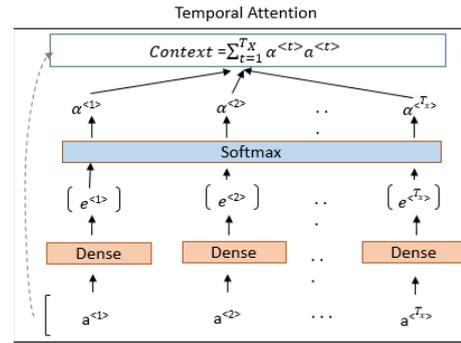


Fig. 3. The temporal attention mechanism.

	Number of trainable parameters	Training time per epoch
CNN	649, 682	22 secs
CNN-LSTM	601, 458	28 secs
CNN-AtLSTM	601, 491	28 secs

Table 1. Details for the models with sample hyper-parameter setting  $T_z = 32$  &  $T_y = 4$

context vector as shown in Fig. 3. It is trained jointly with the network.

#### 3.2 Models

We perform experiments with 3 models, out of which the model 3D CNN is without any LSTM layers. We refer to the ‘3D CNN’ model as the ‘CNN’ now onwards in this paper to avoid clutter. The ‘3D CNN LSTMs’ model and the ‘3D CNN LSTMs Attention’ model are referred to as the ‘CNN-LSTM’ and ‘CNN-AtLSTM’ respectively.

Our CNN model (test accuracy of which is reported) has a slightly different architecture than that of the 3D CNN block illustrated in Fig. 4. The 3D CNN block is only used as part of the other two models as shown in Fig. 5. We develop the CNN model to keep its number of parameters approximately on the same scale as the other two models for ensuring an unbiased comparative study. For example, if we have used the 3D CNN block of Fig. 4 as our model, the number of trainable parameters would have been 4.2 million (For  $T_z = 32$ ,  $T_y = 4$ ) instead of 649, 682 (Table 1). This surge in the number of parameters occurs as the 3D CNN block operates on  $T_y$  frames whereas the CNN model has  $T_z$  frames as input. To keep the parameters in check, we reduce the dimension of the volume before using the fully-connected (FC) block. To achieve this, we modify the 3D CNN block (Fig. 4) using another layer of Conv 3D, Maxpool 3D before the FC layer of 128 hidden units. That becomes our CNN model which also has an extra FC layer (128 units) before the softmax layer.

In the models CNN-LSTM and CNN-AtLSTM, the intuition behind using 3D CNN is that the 3D CNN blocks (each taking in a volumetric image sequence as input at every time-step, Fig. 5) can filter the high-frequency noise prevailing in the dataset. The encoding vectors computed by the 3D CNN are then fed into stacked LSTM layers to explicitly learn the temporal correlations on top of that. This is the advantage compared to models using 2D CNN LSTM (taking in an image input at every time-step before LSTM layers) which are very sensitive to noise and thus

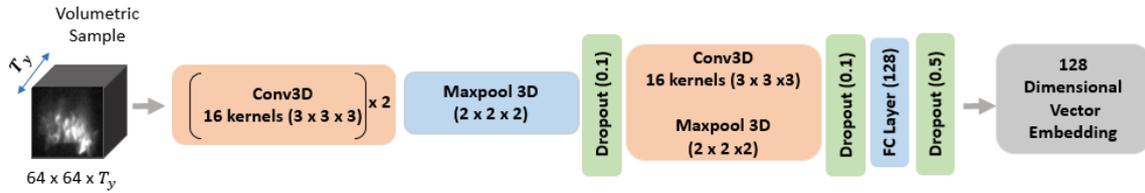


Fig. 4. The details of the 3D CNN block used to encode volumetric image samples ( $T_y$  frames, each frame of dimension  $64 \times 64$ ) into a 128-dimensional vector for the models CNN-LSTM & CNN-AtLSTM.

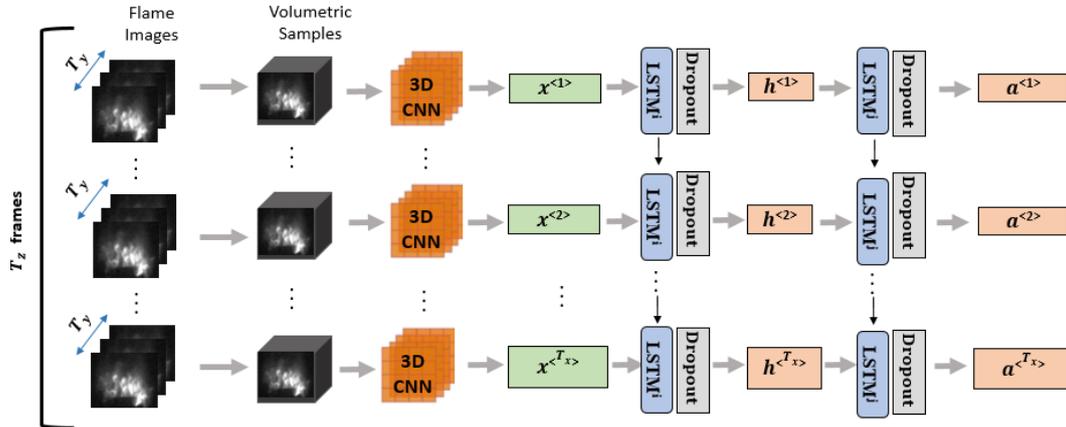


Fig. 5. From a sequence of  $T_z$  frames,  $T_x$  volumetric samples are formed and each volumetric sample consists of  $T_y$  frames. The 3D CNN block (Fig. 4) computes a 128-dimensional embedding for each volumetric sample. The sequence of these embeddings acts as input to the first LSTM layer. The hidden states of the first LSTM layer are fed into the second LSTM layer to get the annotations.

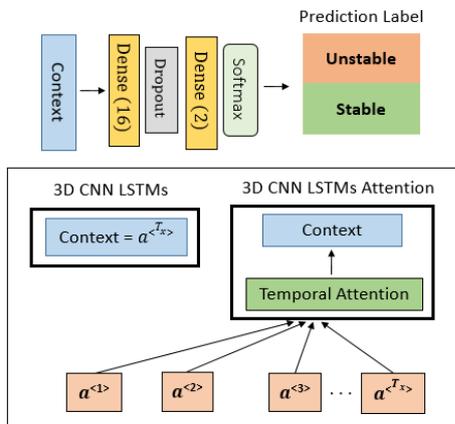


Fig. 6. The CNN-LSTM model (with 3D CNN LSTMs) and the CNN-AtLSTM model (with 3D LSTMs Attention) after the encoding part (common for both)

cannot generalize well leading to inaccurate predictions of the test set. In the CNN-LSTM model (Fig. 6), the last annotation of the encoding part is assumed to be the compressed representation of the entire input sequence. The CNN-AtLSTM model (Fig. 6) takes in all the annotations as input to generate the context vector using the temporal attention mechanism. The hyper-parameters of the models are tuned with different experiments. Compared to the CNN-LSTM model, there is only an increase of 33 learnable parameters for the CNN-AtLSTM Model, which is insignificant compared to the total number of parameters. The training time remains the same. We implement the models using Keras (Chollet et al. (2015)) with the TensorFlow backend (Abadi et al. (2016)). The models are

trained using NVIDIA GPUs. Categorical cross-entropy is used as the loss function. We use Adam optimizer (Kingma and Ba (2014)) with the learning rate of  $3e - 4$ .

#### 4. RESULTS

To demonstrate the robustness of our models, we carefully prepare a dataset that has a balanced number of samples of stable and unstable cases in the training set. We label the dataset using the labeling procedure described in Section 2.2. From Fig. 2, there are more stable conditions than unstable ones. To balance the number of samples between stable and unstable cases in the training dataset, both 4ft\_12\_run\_a and 4ft\_12\_run\_b are used as unstable conditions in the model training. In the test set, we choose a condition from the “2ft Tube” which shows different dynamics and one condition from the “4ft Tube” which is unseen for the model. Table 2 shows the choices of experimental conditions used for the model training and testing process. From Fig. 7, we can see that the 2ft\_3\_run\_a condition generally has the highest K-L Divergence values among other conditions and thus, can be termed as an unstable condition (when referred to Fig. 2). From a domain knowledge perspective, the fundamental frequency of the 2ft tube case is 300 Hz and we use the frequency range of 150 - 450 Hz (+/- 50% of the fundamental frequency) to compute the K-L Divergence for the 2ft tube.

The performance of the models in terms of test accuracy is shown in Table 3. We use a time window of 0.1s which corresponds to 500 images (5000 frames/sec of the hi-speed video). For each time window, the input to our model is a volumetric sample of  $T_z$  frames. The video is, therefore, down-sampled from 500 frames to  $T_z$  frames due

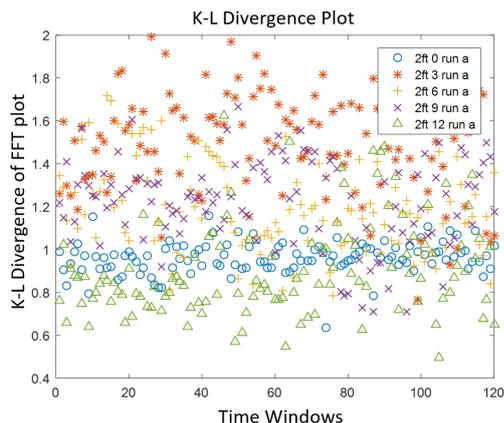


Fig. 7. The K-L Divergence of FFT plots (for each time window) for the 2ft tube relative to the baseline FFT.

	Train	Test
Stable	4ft_0_run_a	4ft_6_run_a
	4ft_3_run_a	
Unstable	4ft_12_run_a	2ft_3_run_a
	4ft_12_run_b	

Table 2. The training and test set chosen for our experiments.

Hyperparameters			Test Accuracy (%)		
$T_z$	$T_y$	$T_x$	CNN	CNN-LSTM	CNN-AtLSTM
32	4	8	100	100	100
32	8	4	100	100	100
48	4	12	100	99.58	100
48	8	6	100	100	100
64	4	16	100	100	99.58
64	8	8	100	100	100

Table 3. Test accuracy of 3 models for different combinations of  $T_z$  and  $T_y$ , where  $T_x = T_z/T_y$

to computation restrictions. We down-sample the video by taking frames after certain intervals (depending on  $T_z$ ). The test set accuracy is almost 100% for all the hyperparameter combinations. This highlights that the models can correctly classify the 2ft\_3\_run\_a as unstable, despite being trained only on data from the 4ft tube. Fig. 7 also confirms that 2ft\_3\_run\_a is actually an unstable condition. Our proposed deep learning models are robust enough to generalize well in detecting instability of an unseen condition showing contrasting dynamics with different dominant frequency modes.

Considering only the accuracy of the model ignores the importance of interpretability in a deep learning model (Varshney et al. (2018)). It is important to consider interpretability for increased safety, better understanding, error identification, and enhancing the confidence in implementing DL models. We validate the insights learned by our model from the domain knowledge perspective to verify the interpretability. The attention weights indicate the most relevant time-periods in generating a particular prediction (Fig. 8). We observe from the results that the model is not only focusing on the first or last few frames arbitrarily. Our validation approach is shown in Fig. 8 for a sample sequence of the 2ft\_3\_run\_a condition. We divide the corresponding acoustic time series (0.1 secs) into  $T_x$  (8 for the example) windowed time series (each of length

0.0125 secs). After computing the maximum amplitudes from the FFT plots of each time window, we scale the amplitudes to have a sum of 1. The maximum amplitude in a time window occurs near the fundamental frequency (300 Hz for the 2ft tube) of the FFT plot and doesn't capture noise. It gives an estimation of the existing instability. The distribution of the attention weights (including the peaks) is almost similar to that of the maximum amplitude (which highlights combustion instability), highlighting that the CNN-AtLSTM model focuses correctly on the most important time-periods.

## 5. CONCLUSIONS

The transition from stable to unstable states in combustion systems can happen quickly and can potentially lead to a disastrous system failure. Hence, an accurate and insightful instability monitoring model can be extremely beneficial to prevent such an event. In this paper, we propose an interpretable deep learning framework using 3D CNN, LSTM, and temporal attention mechanism that not only achieves astounding instability classification accuracy on the hi-speed combustion video but also provides insights behind the predictions. The insights are found to be in accord with domain knowledge. The performance is compared with two other deep learning models using 3D CNN and 3D CNN LSTMs. This interpretable deep learning framework can be used for the development of accurate online detection frameworks in different dynamical systems.

## REFERENCES

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, 265–283.
- Akintayo, A., Lore, K.G., Sarkar, S., and Sarkar, S. (2016). Prognostics of combustion instabilities from hi-speed flame video using a deep convolutional selective autoencoder. *International Journal of Prognostics and Health Management*, 7(023), 1–14.
- Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Bengio, Y. (1991). *Artificial neural networks and their application to sequence recognition*. McGill University.
- Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2), 157–166.
- Berkooz, G., Holmes, P., and Lumley, J.L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1), 539–575.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- Chollet, F. et al. (2015). Keras.
- Dowling, A.P. (1997). Nonlinear self-excited oscillations of a ducted flame. *Journal of fluid mechanics*, 346, 271–290.

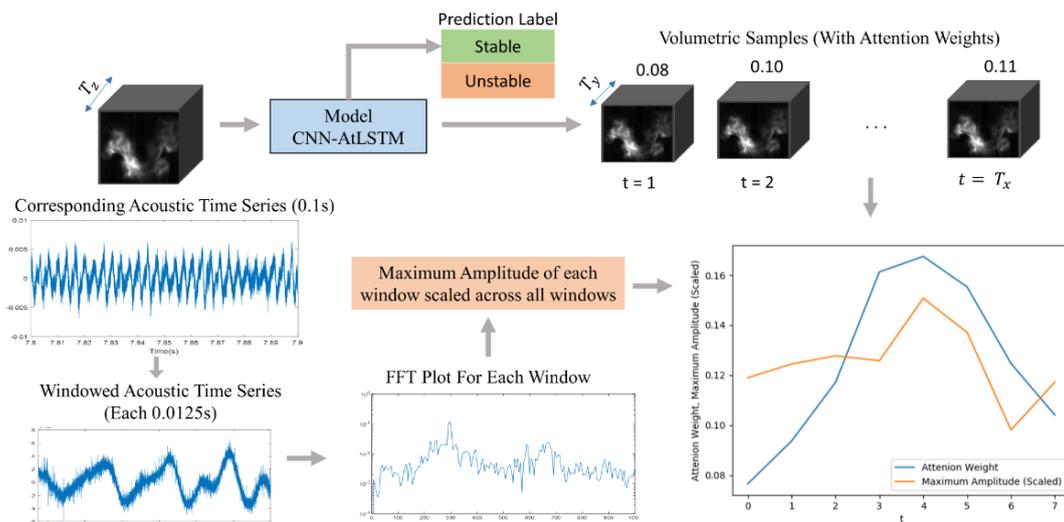


Fig. 8. The insights learned by the CNN-AtLSTM model validated from a domain knowledge perspective.

- Fisher, S.C. and Rahman, S.A. (2009). Remembering the giants: Apollo rocket propulsion development.
- Gangopadhyay, T. (2019). Deep learning for monitoring cyber-physical systems.
- Gangopadhyay, T., Locurto, A., Boor, P., Michael, J.B., and Sarkar, S. (2018). Characterizing combustion instability using deep convolutional neural network. In *ASME 2018 Dynamic Systems and Control Conference*, V001T13A004–V001T13A004. American Society of Mechanical Engineers.
- Gangopadhyay, T., Locurto, A., Michael, J.B., and Sarkar, S. (2020). Deep learning algorithms for detecting combustion instabilities. In *Dynamics and Control of Energy Systems*, 283–300. Springer.
- Gers, F.A., Schmidhuber, J., and Cummins, F. (1999). Learning to forget: Continual prediction with lstm.
- Ghosal, S., Akintayo, A., Boor, P., and Sarkar, S. (2017). High speed video-based health monitoring using 3d deep learning. *Dynamic Data-Driven Application Systems (DDDAS)*.
- Ghosal, S., Ramanan, V., Sarkar, S., Chakravarthy, S.R., and Sarkar, S. (2016). Detection and analysis of combustion instability from hi-speed flame images using dynamic mode decomposition. In *ASME 2016 Dynamic Systems and Control Conference*, V001T12A005–V001T12A005. American Society of Mechanical Engineers.
- Hinton, G.E. and Salakhutdinov, R.R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786), 504–507.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.
- Kingma, D.P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- LoCurto, A., Gangopadhyay, T., Boor, P., Sarkar, S., and Michael, J.B. (2018). Mode decomposition and convolutional neural network analysis of thermoacoustic instabilities in a rijke tube.
- Nair, V. and Sujith, R. (2014). Multifractality in combustion noise: predicting an impending combustion instability. *Journal of Fluid Mechanics*, 747, 635–655.
- Palies, P., Schuller, T., Durox, D., and Candel, S. (2011). Modeling of premixed swirling flames transfer functions. *Proceedings of the combustion institute*, 33(2), 2967–2974.
- Rayleigh, J.W.S. (1878). The explanation of certain acoustical phenomena. *Nature*, 18(455), 319–321.
- Sarkar, S., Lore, K.G., and Sarkar, S. (2015). Early detection of combustion instability by neural-symbolic analysis on hi-speed video. In *Proceedings of the 2015th International Conference on Cognitive Computation: Integrating Neural and Symbolic Approaches-Volume 1583*, 93–101. CEUR-WS. org.
- Schmid, P.J. (2010). Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656, 5–28.
- Sen, U., Gangopadhyay, T., Bhattacharya, C., Misra, A., Karmakar, S., Sengupta, P., Mukhopadhyay, A., and Sen, S. (2016). Investigation of ducted inverse nonpremixed flame using dynamic systems approach. In *ASME Turbo Expo 2016: Turbomachinery Technical Conference and Exposition*, V04BT04A059–V04BT04A059. American Society of Mechanical Engineers.
- Sen, U., Gangopadhyay, T., Bhattacharya, C., Mukhopadhyay, A., and Sen, S. (2018). Dynamic characterization of a ducted inverse diffusion flame using recurrence analysis. *Combustion Science and Technology*, 190(1), 32–56.
- Sutskever, I., Vinyals, O., and Le, Q.V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 3104–3112.
- Varshney, K.R., Khanduri, P., Sharma, P., Zhang, S., and Varshney, P.K. (2018). Why interpretability in machine learning? an answer using distributed detection and data fusion theory. *arXiv preprint arXiv:1806.09710*.