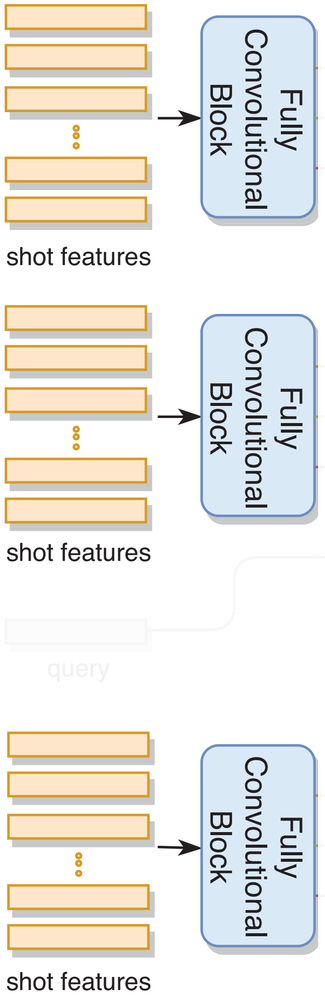
Proposed Method

Fully Convolutional Block

- Use 1D fully-convolutional network architecture to encode the shot-level visual feature
- Utilize **dilated convolutions** to obtain larger receptive field for handling long distance among the video segment



- First propose convolutional networks with different filter size and then concatenate their outputs which enables the model to receive more information
- The dilated convolution operation on i-th shot in a video segment: $o_i = \sum_{t=-k}^{k} f(t) \cdot v_{i+d \cdot t}$, where $2k+1$ is the filter size, f is the filter and d is dilation factor
- Then employ a pooling layer on the temporal axis of the video, can reduce computing time and also decrease the running memory of the model
- Connect different fully convolutional block and construct a multi-layer block to extracted representative features

Proposed Method

Information Fusion Layer

- Denote the output features of fully convolutional block as $(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_t)$
  - t: length of output feature sequence
- **Input**: features from fully convolutional block
- **Output**: Sequence of concatenated vectors
  - {outputs from previous block, local attentive, query-aware global attentive}