CNN for Text Data

CNN for Image Data

Image Classification

CNN for Image Data

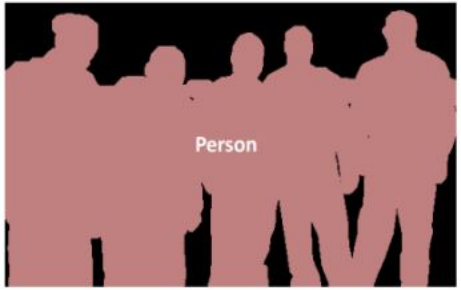
Image Classification

Object Detection

CNN for Image Data

Image Classification

Object Detection

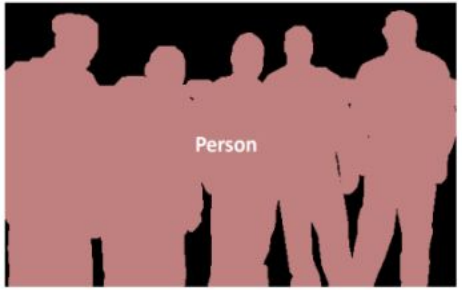


Semantic Segmentation

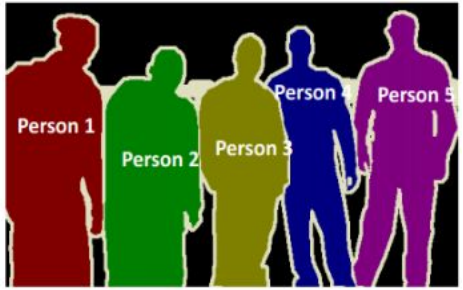
CNN for Image Data

Image Classification

Object Detection



Semantic Segmentation



Instance Segmentation

CNN for Text Data

A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification

Ye Zhang
Dept. of Computer Science
University of Texas at Austin
yezhang@utexas.edu

Byron C. Wallace
iSchool
University of Texas at Austin
byron.wallace@utexas.edu

Abstract

Convolutional Neural Networks (CNNs) have recently achieved remarkably strong performance on the practically important task of sentence classification (Kim, 2014; Kalchbrenner et al., 2014; Johnson and Zhang, 2014). However, these models require practitioners to specify an exact model architecture and set accompanying hyperparameters, including the filter region size, regularization parameters,

comprising each sentence into a vector, forming a matrix to be used as input (e.g., see Fig. 1). The models need not be complex to realize strong results: Kim (2014), for example, proposed a simple one-layer CNN that achieved state-of-the-art (or comparable) results across several datasets. The very strong results achieved with this comparatively simple CNN architecture suggest that it may serve as a drop-in replacement for well-established baseline models, such as SVM (Joachims, 1998) or logistic regression. While more complex deep

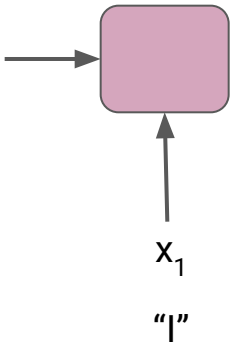
CNN for Text Data

“I like this movie very much!”

CNN for Text Data

"I like this movie very much!"

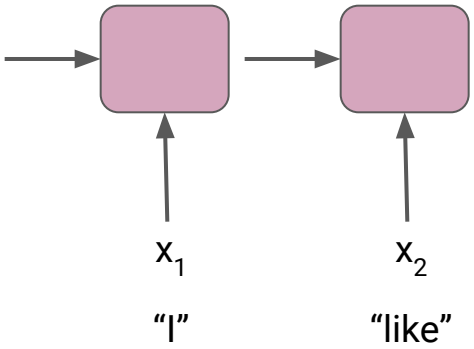
- In RNN/LSTM:



CNN for Text Data

"I like this movie very much!"

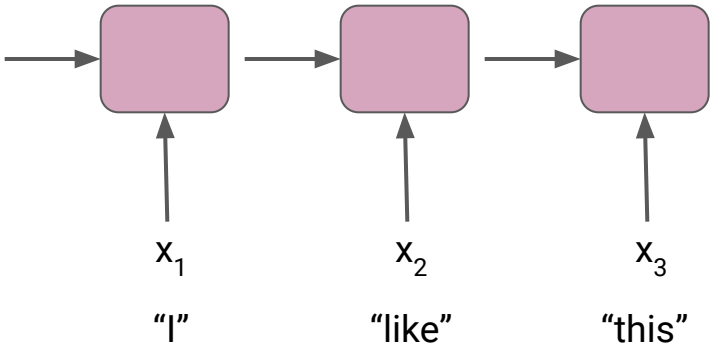
- In RNN/LSTM:



CNN for Text Data

"I like this movie very much!"

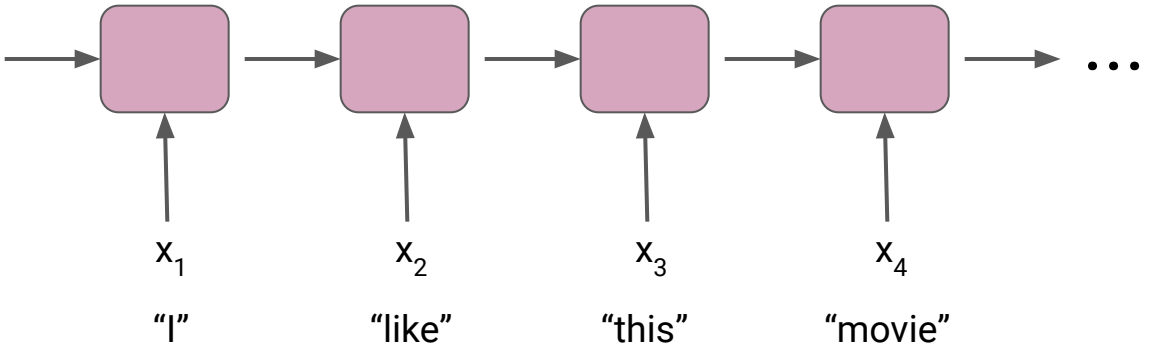
- In RNN/LSTM:



CNN for Text Data

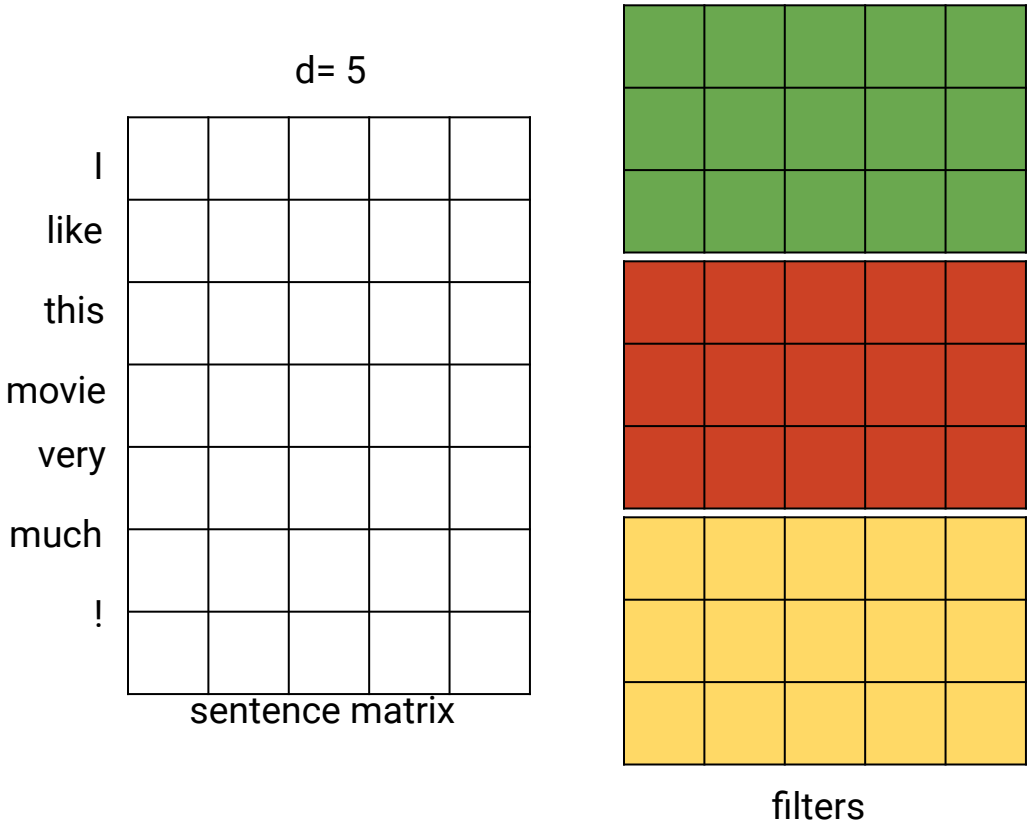
"I like this movie very much!"

- In RNN/LSTM:



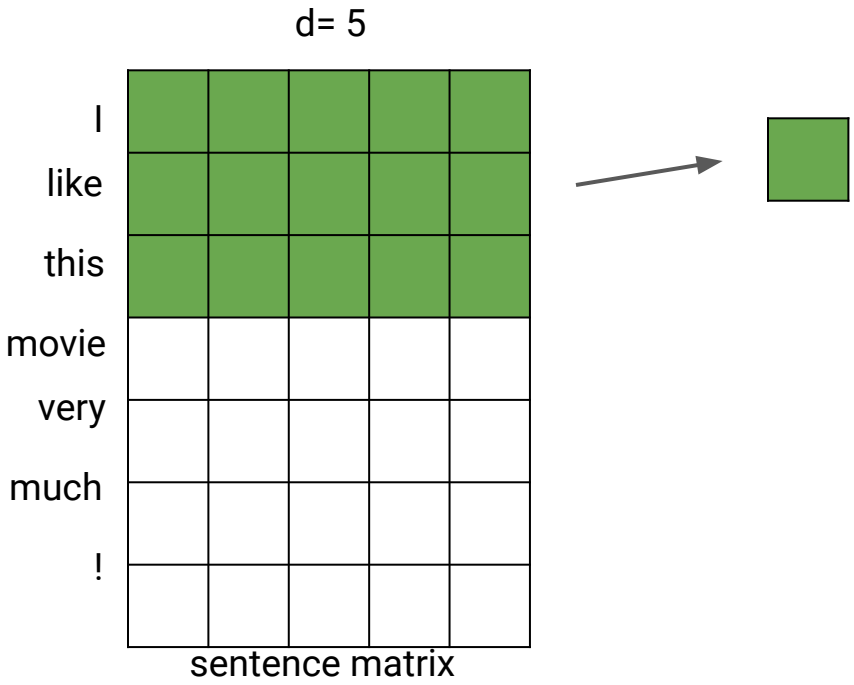
CNN for Text Data

CNN for Text Data

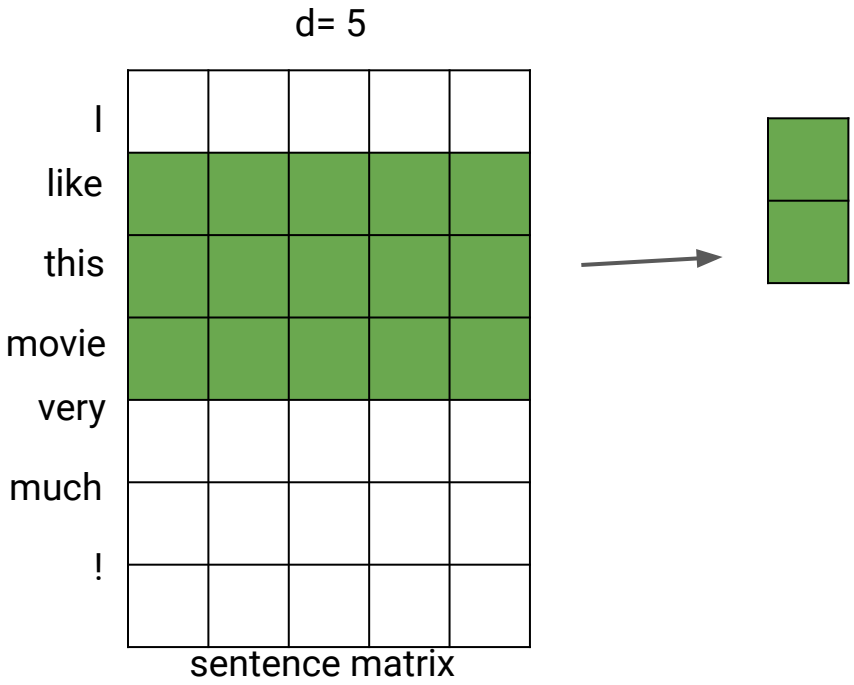


Applying Convolution

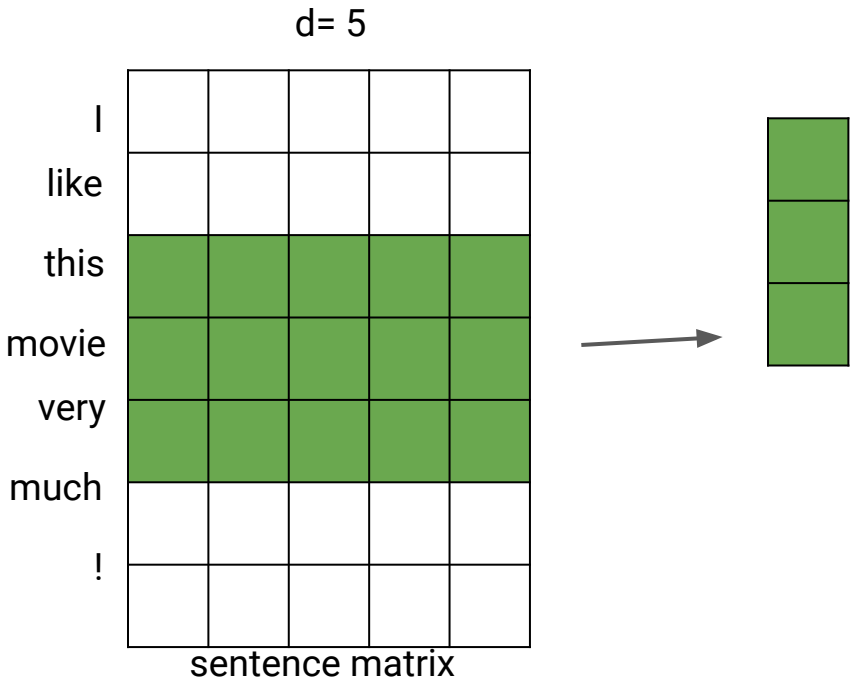
Applying Convolution



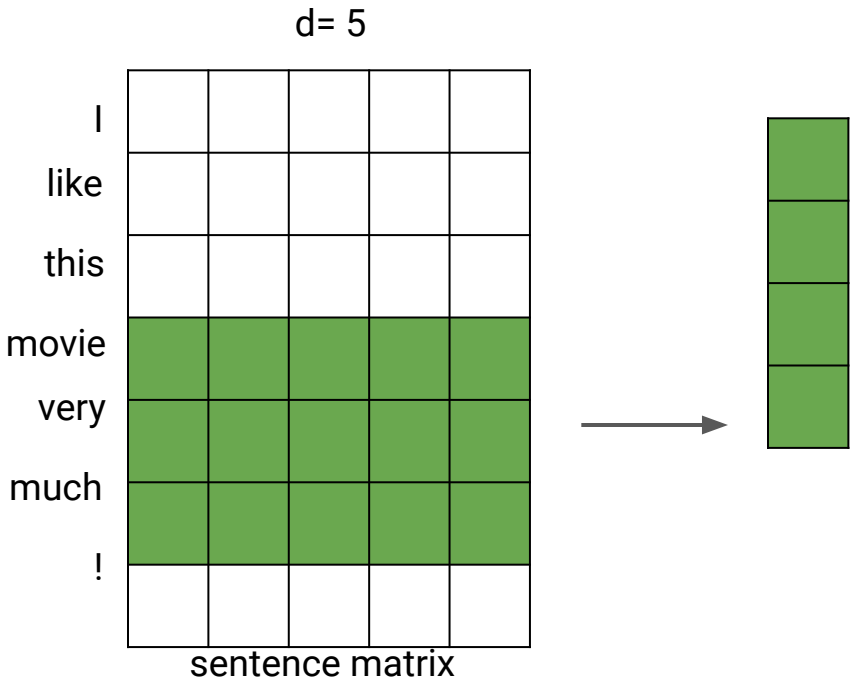
Applying Convolution



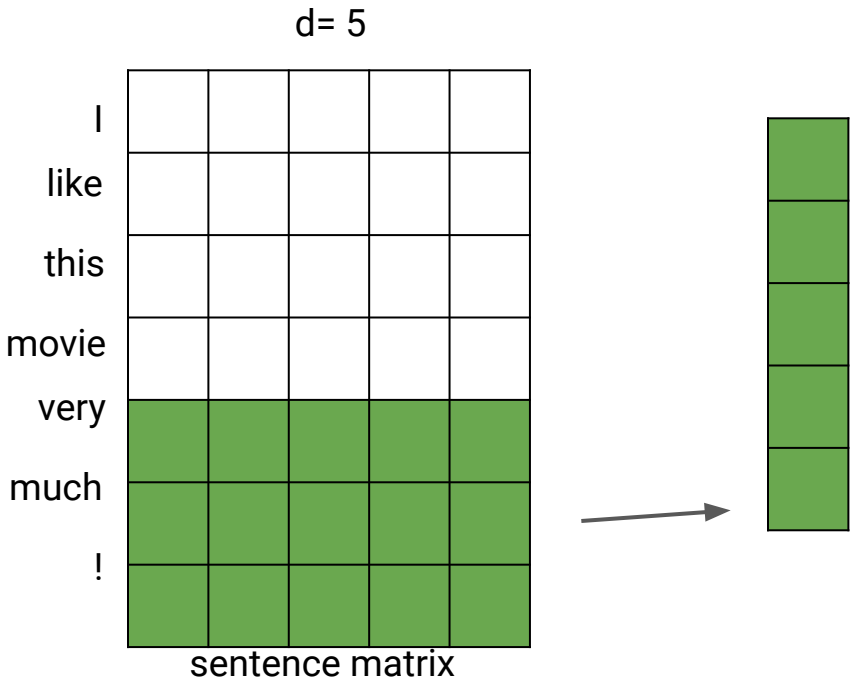
Applying Convolution



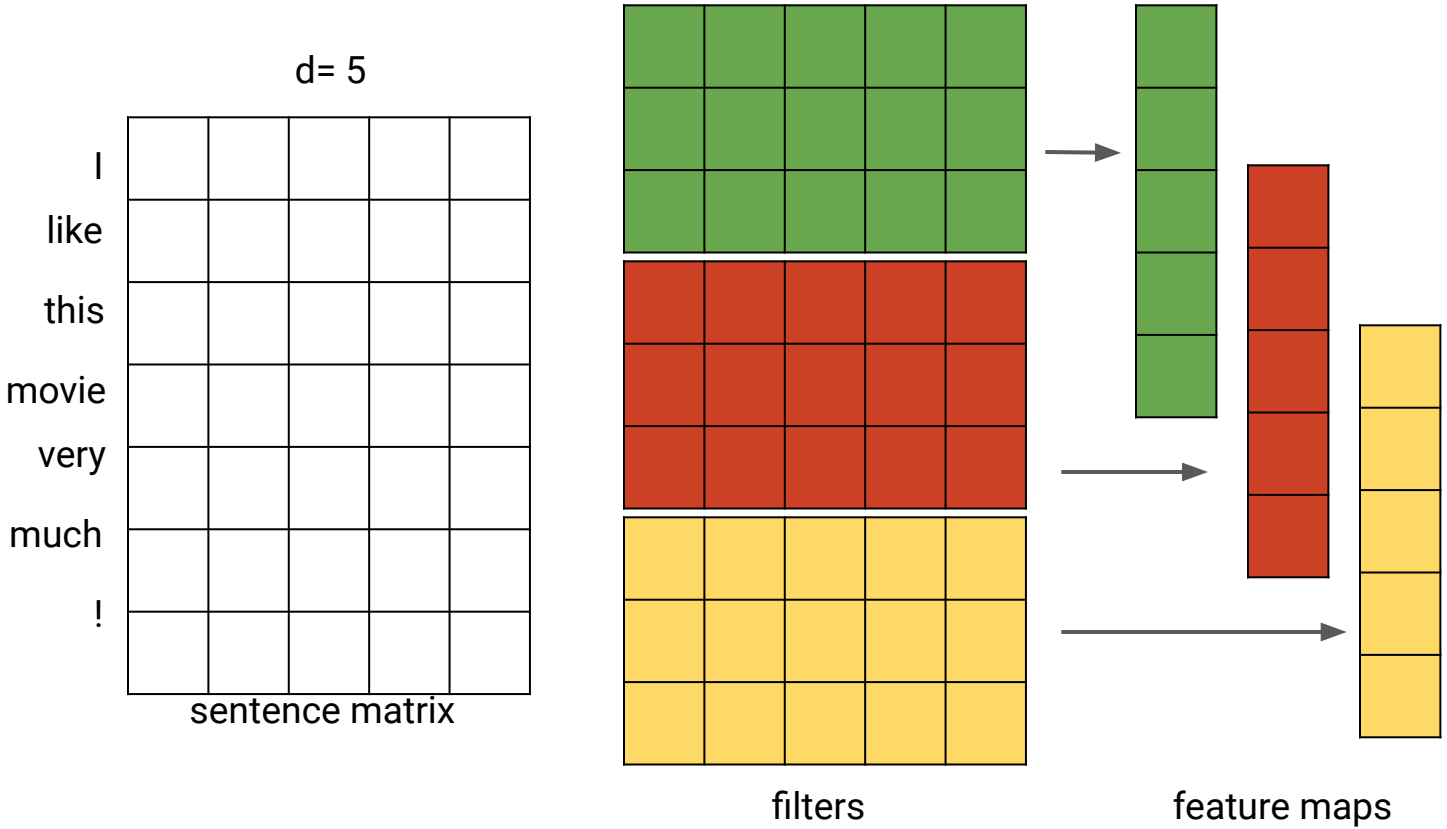
Applying Convolution



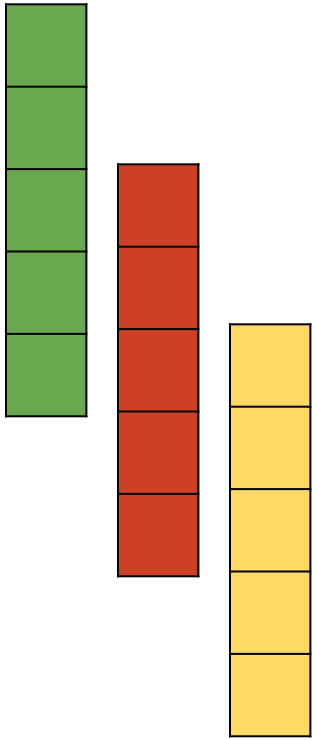
Applying Convolution



CNN for Text Data

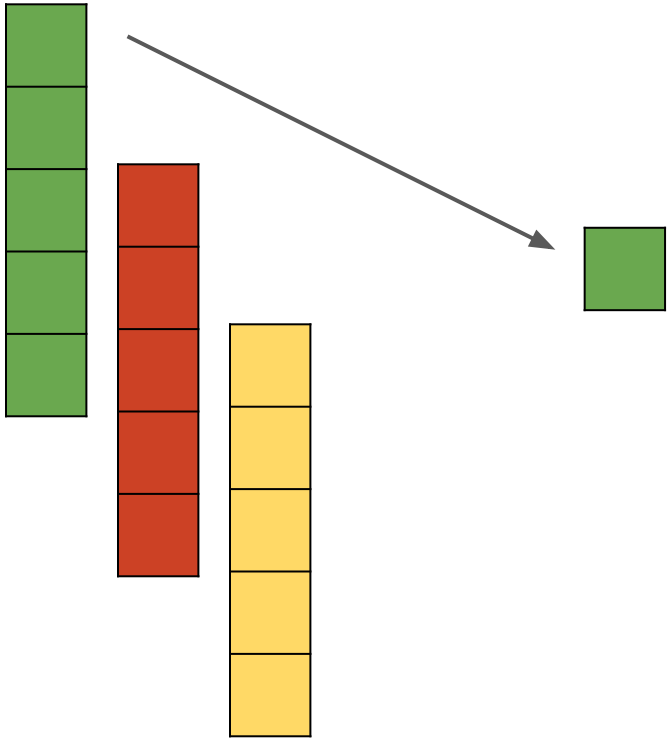


Max Pooling



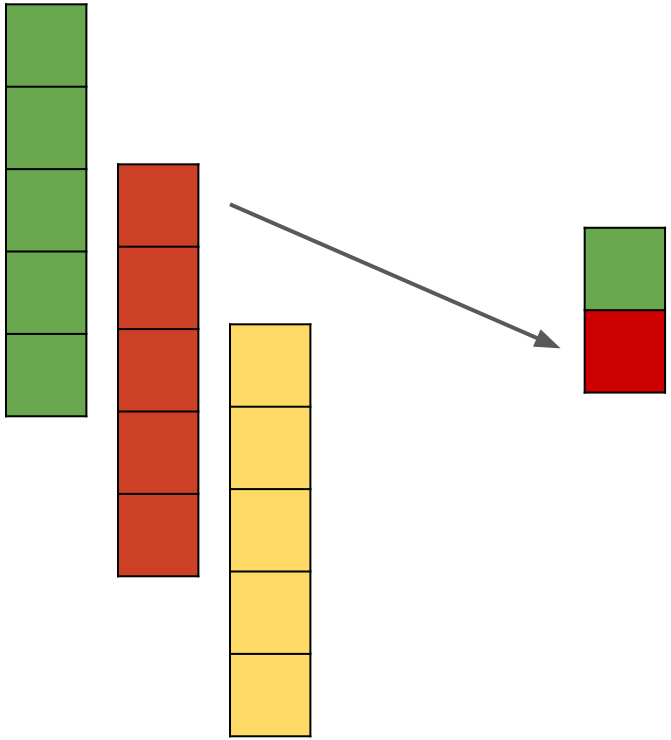
feature maps

Max Pooling



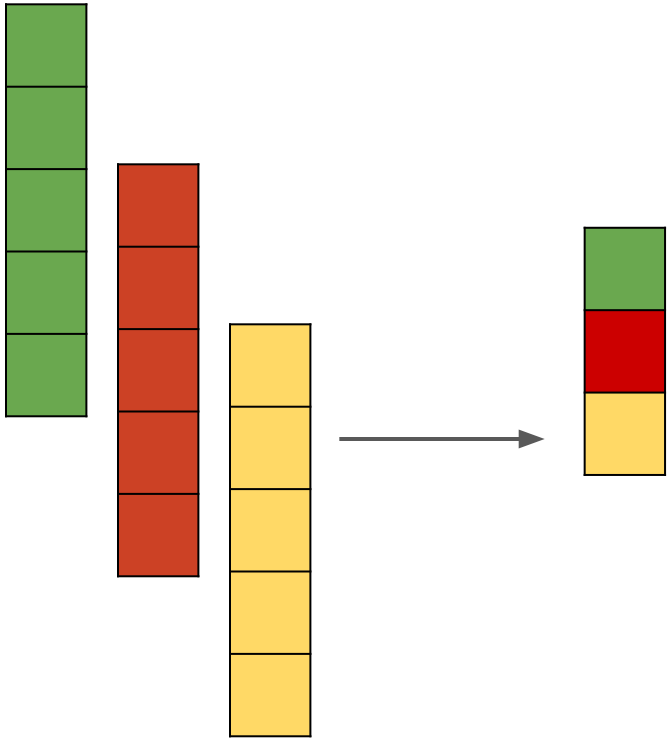
feature maps

Max Pooling



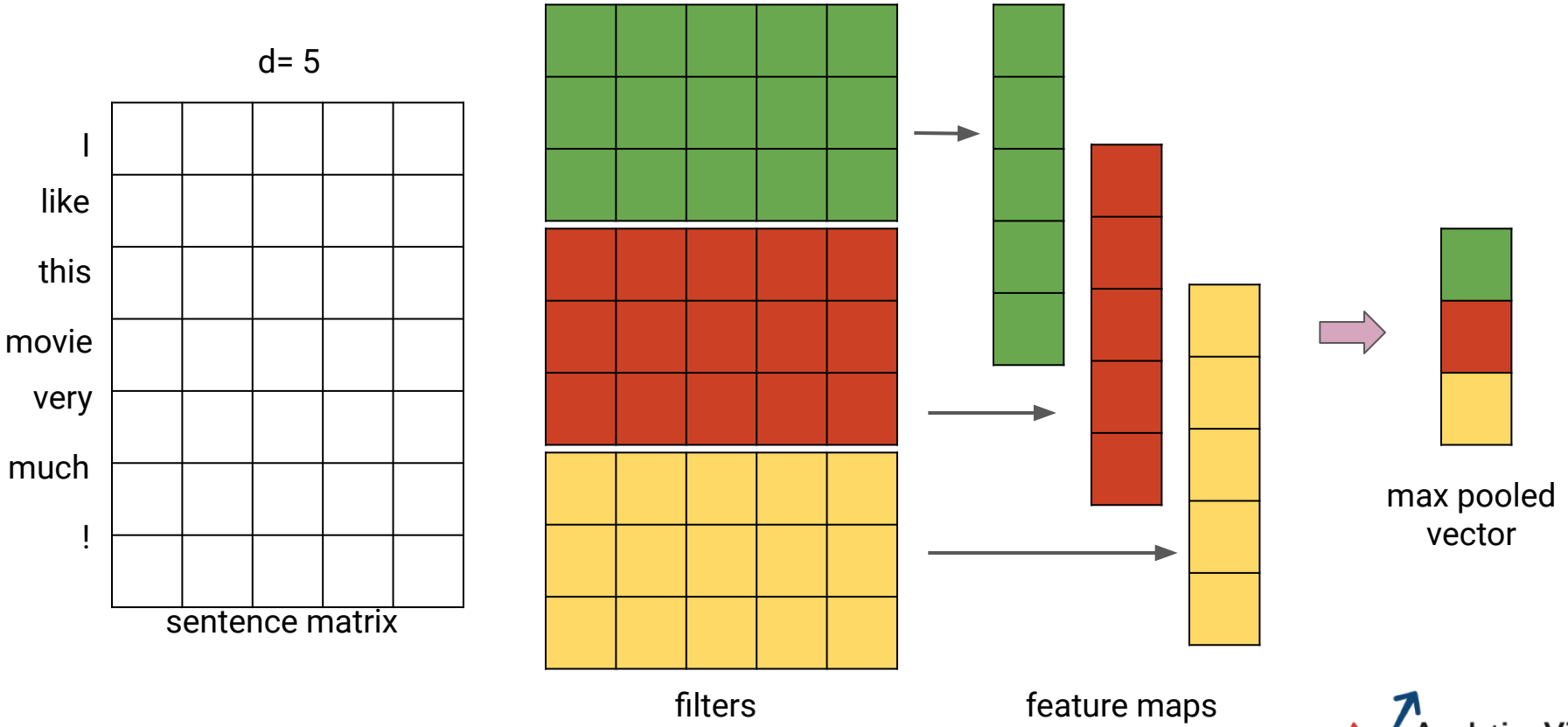
feature maps

Max Pooling



feature maps

CNN for Text Data: Architecture



Thank You