CS224n : NLP with Deep learning

Apr. 14. 2021
NYU Shanghai

Lecture 11 : convolutional network for NLP

- RNN can not capture without prefix context, often catch too much of last word in the final vector

Main CNN/ConvNet idea: compute vectors for every possible word subsequence of a certain length

1d discrete convolution: $(f*g)[n] = \sum_{m=-M}^{M} f[n-m]\, g[m]$

Convolution is classically used to extract features from images.
use a filter or kernel

max-pooling: related to activation cell (works better)
(average-pooling、min-pooling)

- other notion: • stride: jump steps in convolution, compact the representation and get shorter size
  - local max pool: take a few row and take a max pool on just these few rows
  - k-max pooling over time: find K highest value in the column
  - dilation: convolution of convolution result. (layer by layer)

- Paper: Single layer CNN for sentence Classification

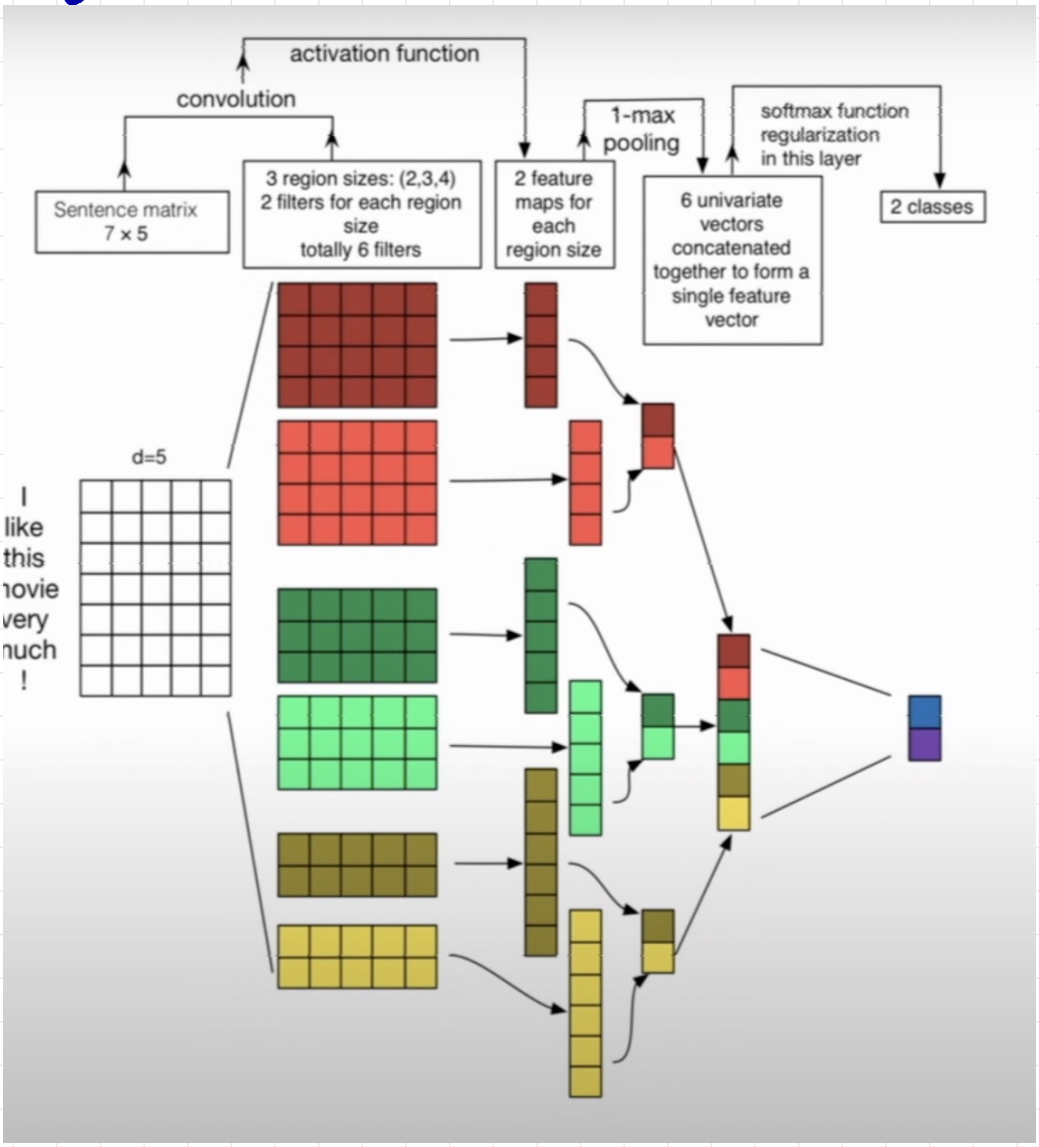
Goal: sentence classification

feature (one channe):

$$c_i = f(w^T x_{i:i+h-1} + b)$$

$$c = [c_1, c_2, \cdots, c_{n-h+1}] \in \mathbb{R}^{n-h-1}$$

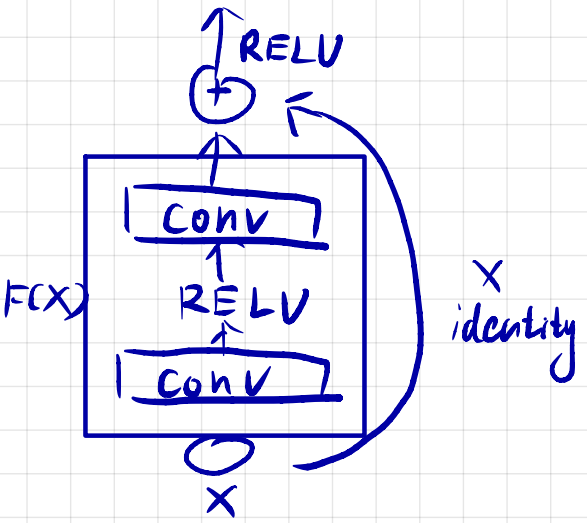
- start with two copies
- backprop in one ; keep the other one static
- max pooling with concentation into one vector



- regularization : • dropout : masking vector r of Bernoulli RV
  • constrain the L2-norm
- Model Comparison: Toolkit
  - Bag of vectors :
  - Window Model :
  - CNN
  - RNN
- Summing candidate update with shortcut connection (Gated unit vertically)

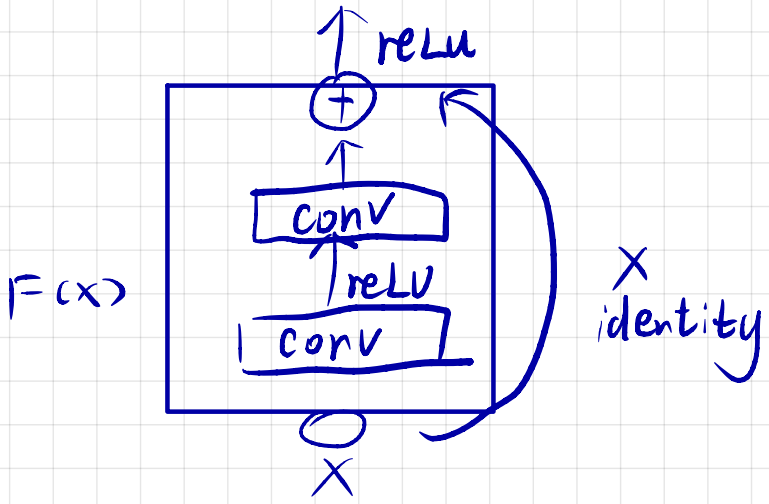
Residual block

$F(x)+x$



Highway block

$F(x)T(x) + x \cdot C(x)$



- Batch normalization
  - transform the output by rescaling to mean=0 var=1 each layer by layer
  - proven effective. Make the initialization less sentitive
- CNN translation

  encoder : CNN
  decoder : RNN

  "Recurrent Continuous Translation Models"
- convolution over characters to generate word embeddings
- Fixed window
- RNN are slow
  - very standard building block for deep learning
  - Idea: take the best and parallelize parts of RNN&CNN

Lecture 12: Subword Models: Information from parts of words

1. Phonetics and phonology

- Morphology: traditionally, morphemes as the smallest semantic unit

Words in writing system vary: • No word segmentation
• Words (mainly) segmented

Models below the word level
- needs to handle large, open vocabulary
- Transliteration
- Informal spelling

1. Word embedding can be composed from character embedding

2. Connected language can be processed as characters

Most deep learning NLP work begins with language in its written form

But human language writing systems aren't one thing!

Purely character-level NMT models (English-Czech)

3. Sub-word models: 2 trend

- Same architecture as for word-level model:
  - use smaller unit "word pieces"
- Hybrid architectures

Byte Pair Encoding (BPE)

- originally a compression algorithm
  - most frequent byte pair → a new byte

replace bytes with character ngrams

- a word segmentation algorithm:
  - start with a vocabulary of characters
  - Most frequent ngram pairs → a new ngram
- Bert use a variant of the wordpiece model