CNNs in Natural Language Processing

Guanlin Li

Nov. 9

Linguistic Aspects That Has Been Applied To

- Sentential
  - Sentence pairs
- Semantic units
  - Events, semantic slots
- Any structures with annotation
- Discourse, text, document

Tasks That Has Been Applied To

- Classification
  - Sentiment
  - Text topic categorization
  - Entailment identification
  - Discourse relation classification
  - KBQA
- Sequence level or labeling
  - Language modelling
  - Parsing

Language Modelling

- Convolutional neural network language models, EMNLP 2016
- A convolutional architecture for word sequence prediction, ACL 2015

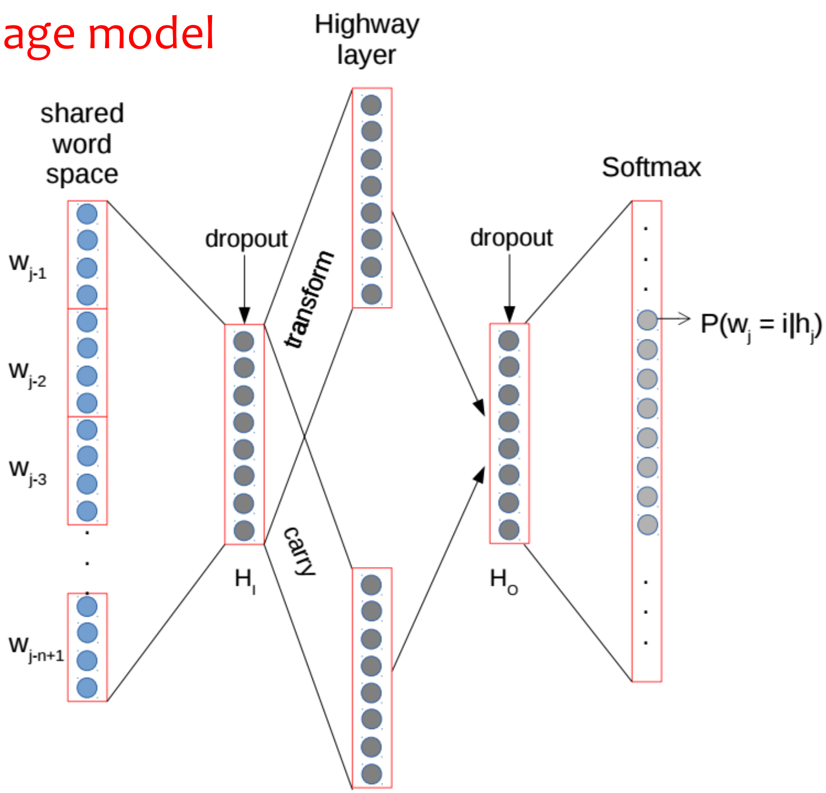
Language Modelling

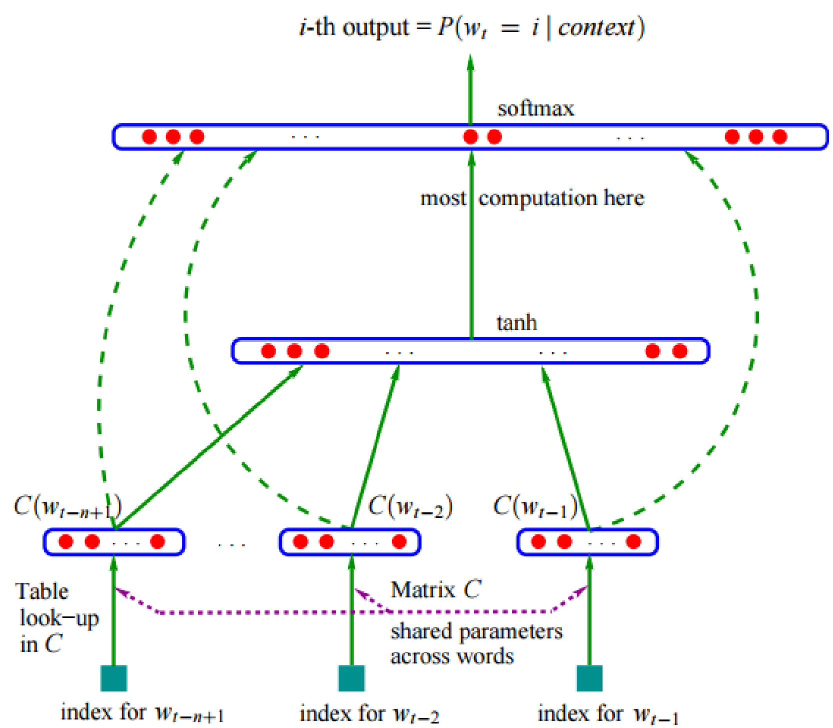
- **Convolutional neural network language models, EMNLP 2016**
  - Language modelling better than FFNN
  - Capture both local & long-range dependency

Language Modelling

- FFNN with Highway layer

N-gram
language model

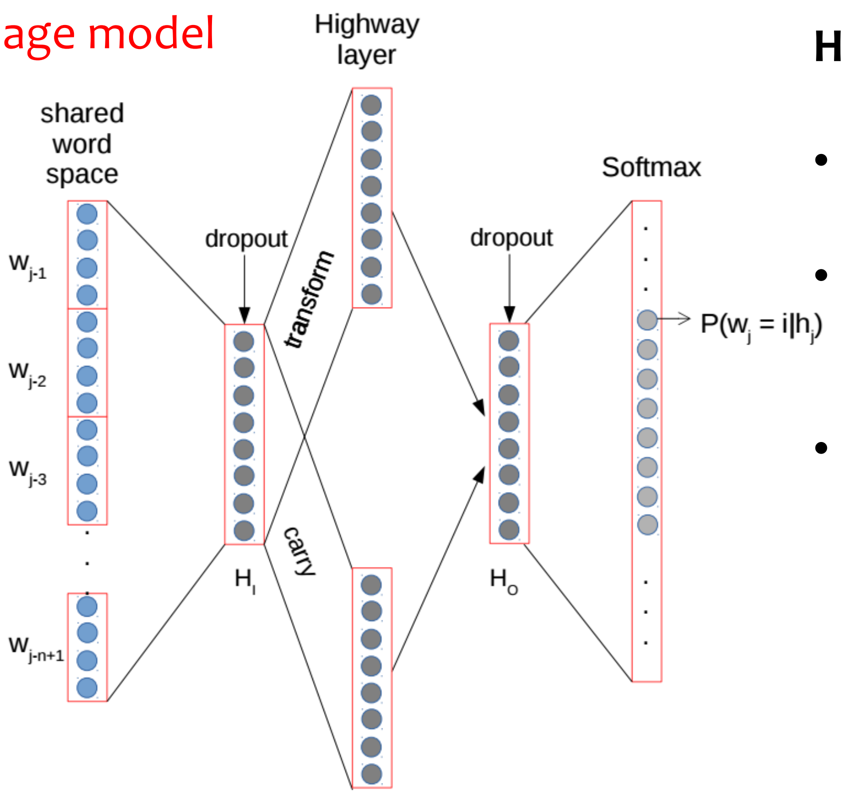




Language Modelling

- FFNN with Highway layer

N-gram
language model

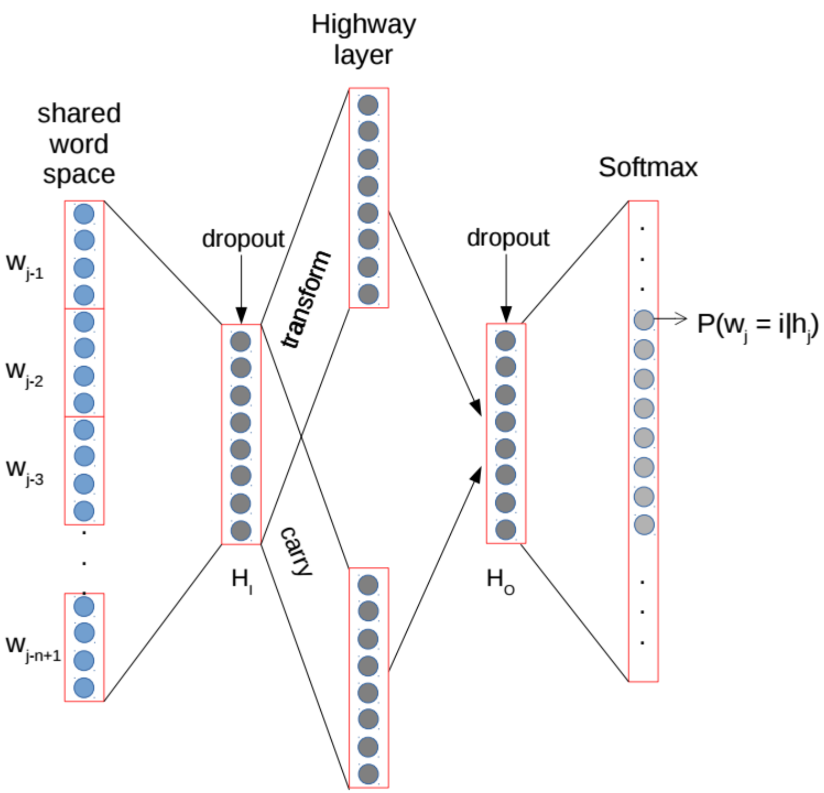


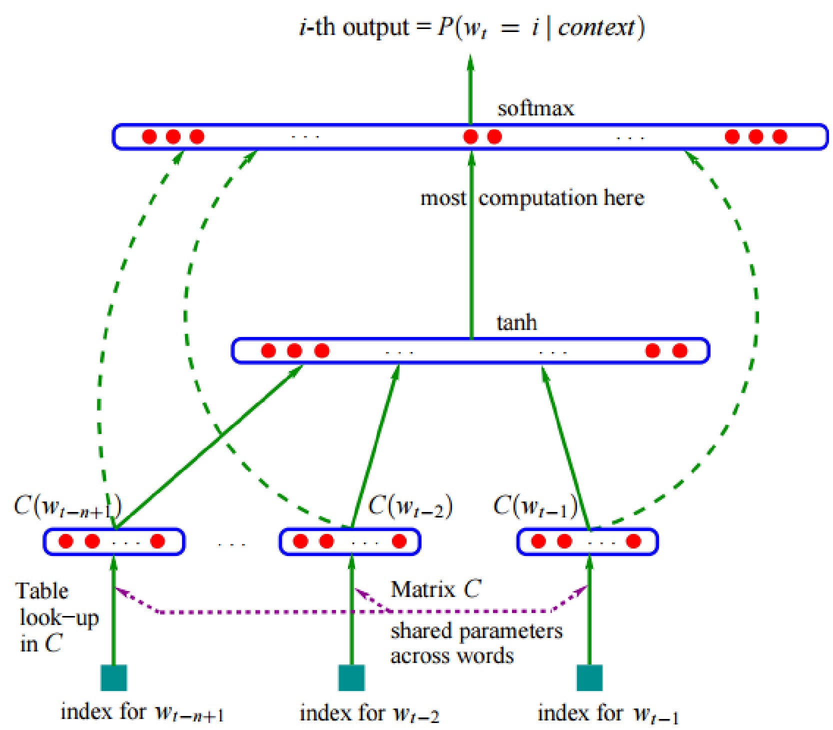
Highway layer

- $H_O = \boldsymbol{g} \odot H_I + (\mathbf{1} - \boldsymbol{g}) \odot tanh(WH_I + b)$
- Where $\boldsymbol{g}$ is a gating vector, that controls information flow
- $\boldsymbol{g}$ is computed by $\sigma(W_T H_I + b_T)$

Language Modelling

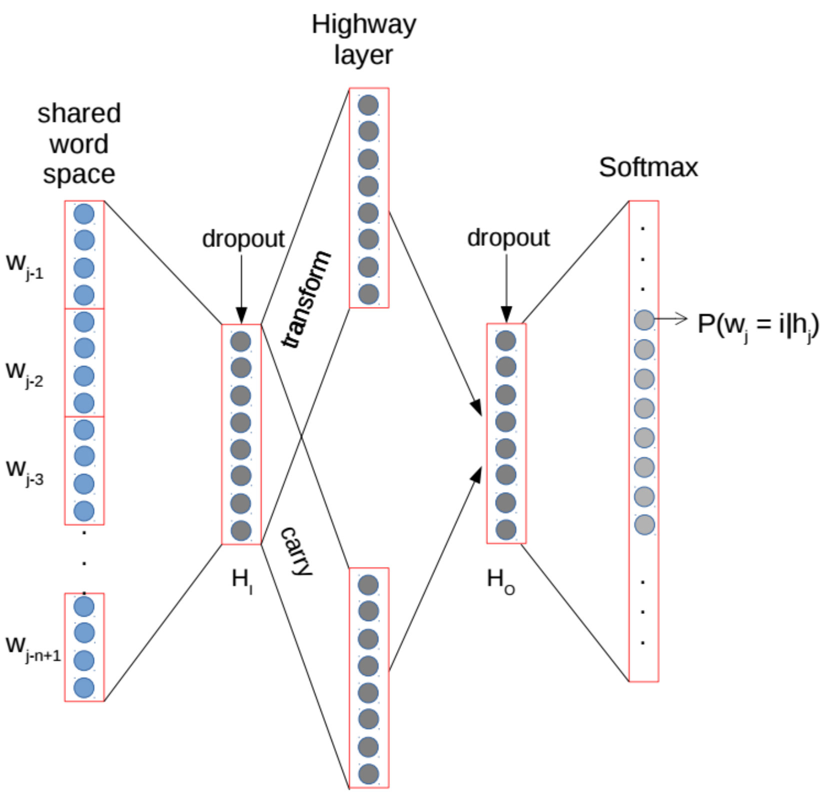
- Compared with Bengio 03

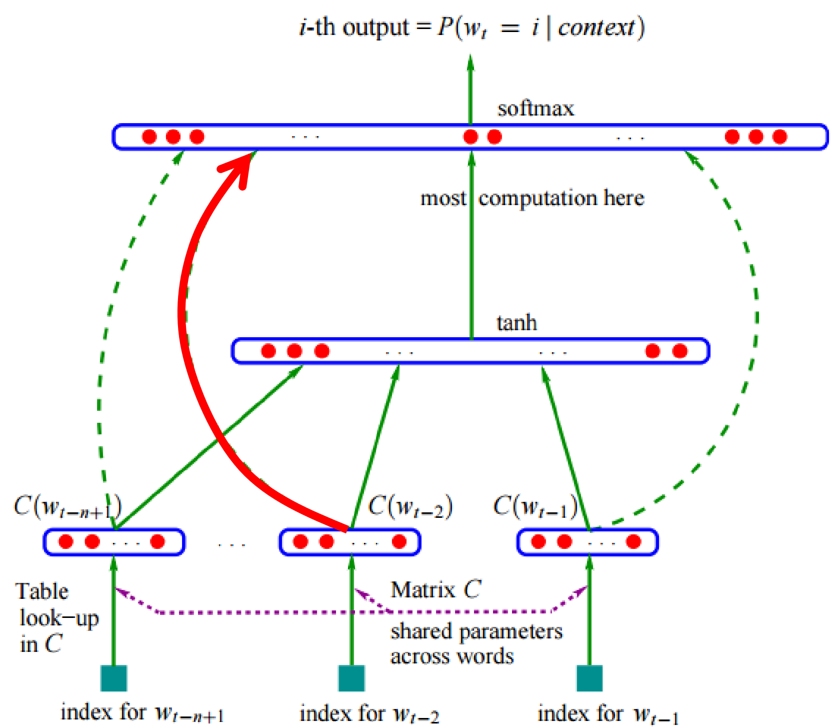




Language Modelling

- Compared with Bengio 03



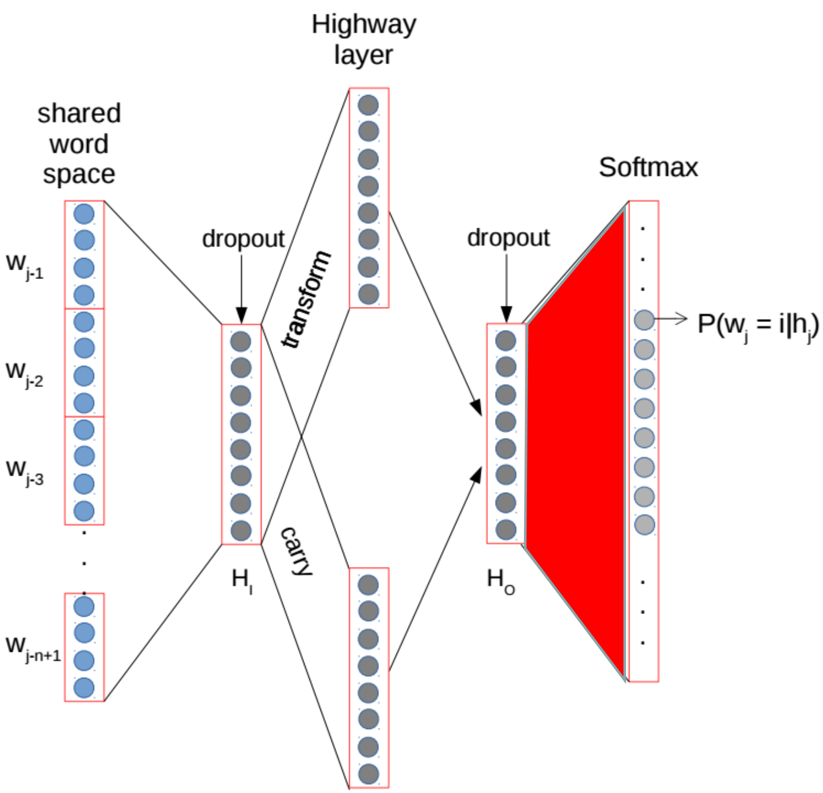


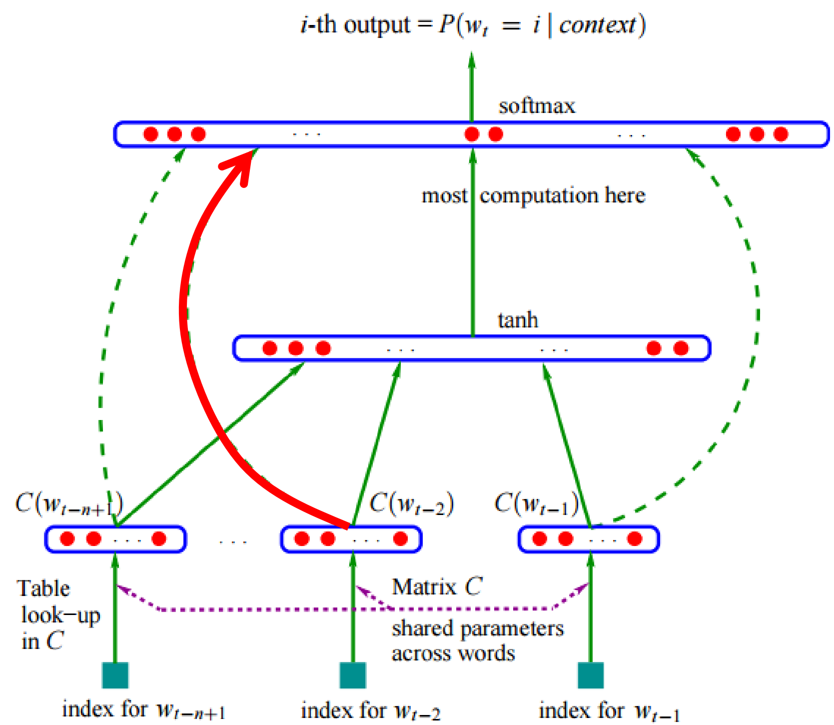
$$H_o = \boldsymbol{g} \odot H_I + (\boldsymbol{1} - \boldsymbol{g}) \odot tanh(WH_I + b)$$

$$\boldsymbol{y} = \boldsymbol{x} + \tanh(W\boldsymbol{x} + \boldsymbol{b})$$

Language Modelling

- Compared with Bengio 03



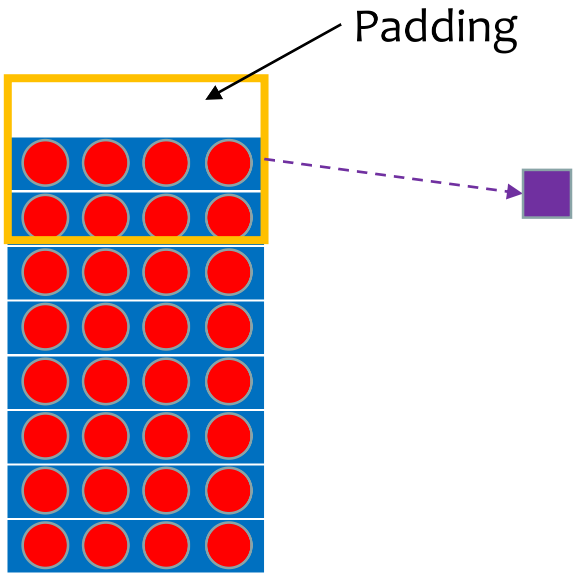


$$H_o = \boldsymbol{g} \odot H_I + (\mathbf{1} - \boldsymbol{g}) \odot tanh(WH_I + b)$$

$$\boldsymbol{y} = \boldsymbol{x} + \tanh(W\boldsymbol{x} + \boldsymbol{b})$$

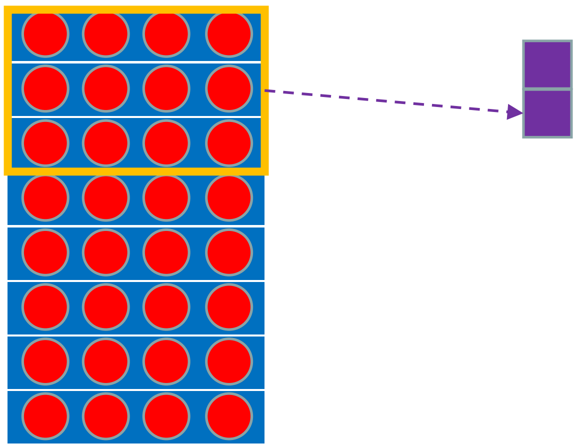
Language Modelling

- The Basic CNN Model



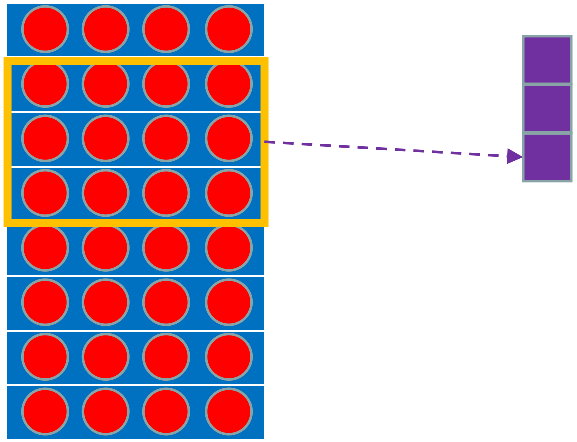
Language Modelling

- The Basic CNN Model



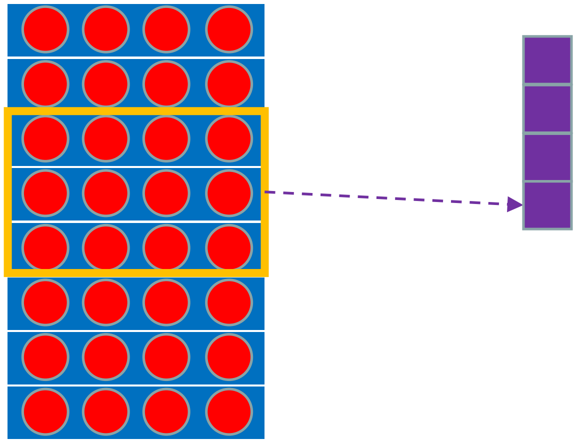
Language Modelling

- The Basic CNN Model



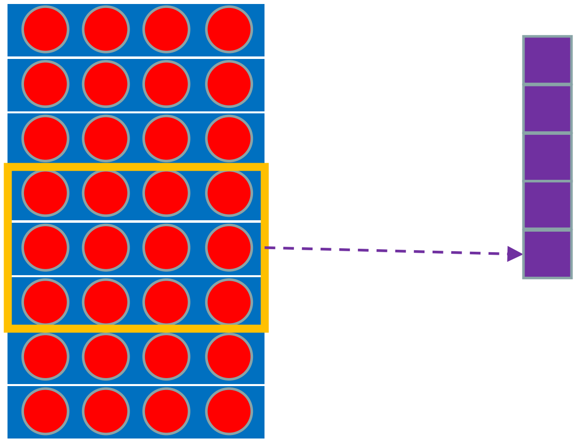
Language Modelling

- The Basic CNN Model



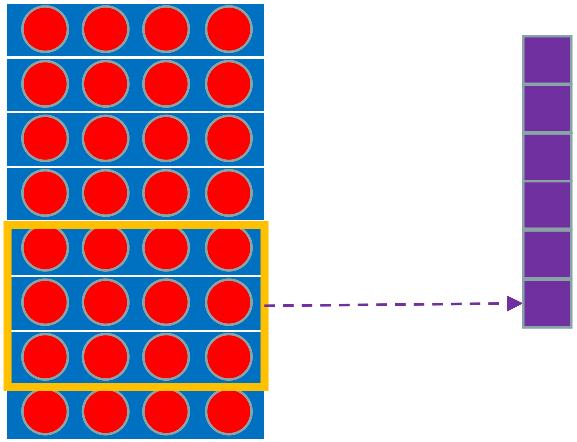
Language Modelling

- The Basic CNN Model



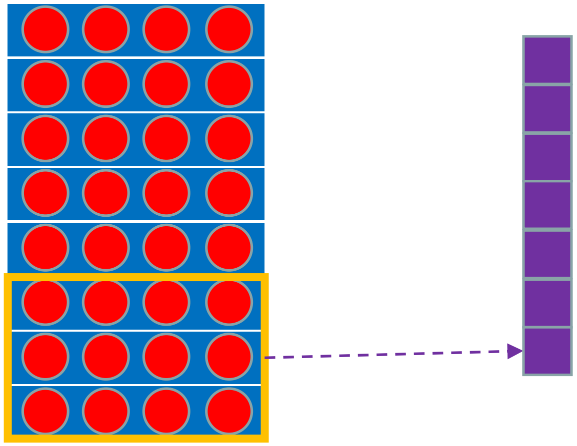
Language Modelling

- The Basic CNN Model



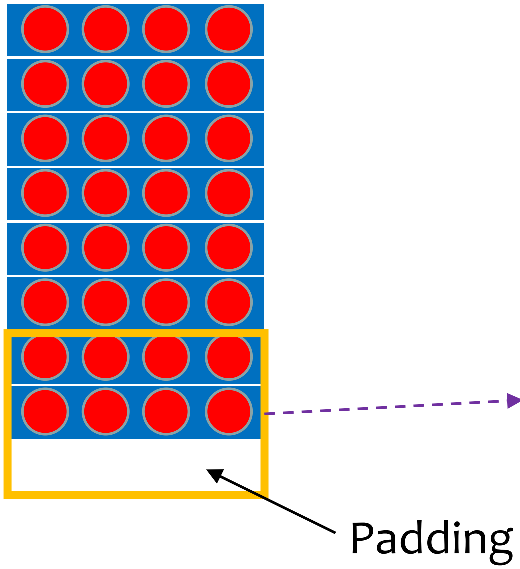
Language Modelling

- The Basic CNN Model



Language Modelling

- The Basic CNN Model



$W \in \mathbb{R}^{w \times k}$

Make x_i as the center, take around w words $x_{i-\frac{w}{2}:i+\frac{w}{2}} \in \mathbb{R}^{w \times k}$ for convolution

Language Modelling

- **A convolutional architecture for word sequence prediction, ACL 2015**

Language Modelling

- **A convolutional architecture for word sequence prediction, ACL 2015**

Sentence Modelling

- Convolutional neural networks for sentence classification, EMNLP 2014
- MGNC-CNN: a simple approach to exploiting multiple word embeddings for sentence classification, ACL 2016
- A convolutional neural network for modelling sentences, ACL 2014

Sentence Modelling

- **Convolutional neural networks for sentence classification, EMNLP 2014**
  - Sentiment analysis (prediction)
  - Question classification

- "In the present work, we train a simple CNN with one layer of convolution on top of word vectors obtained from an unsupervised neural language model."

Sentence Modelling

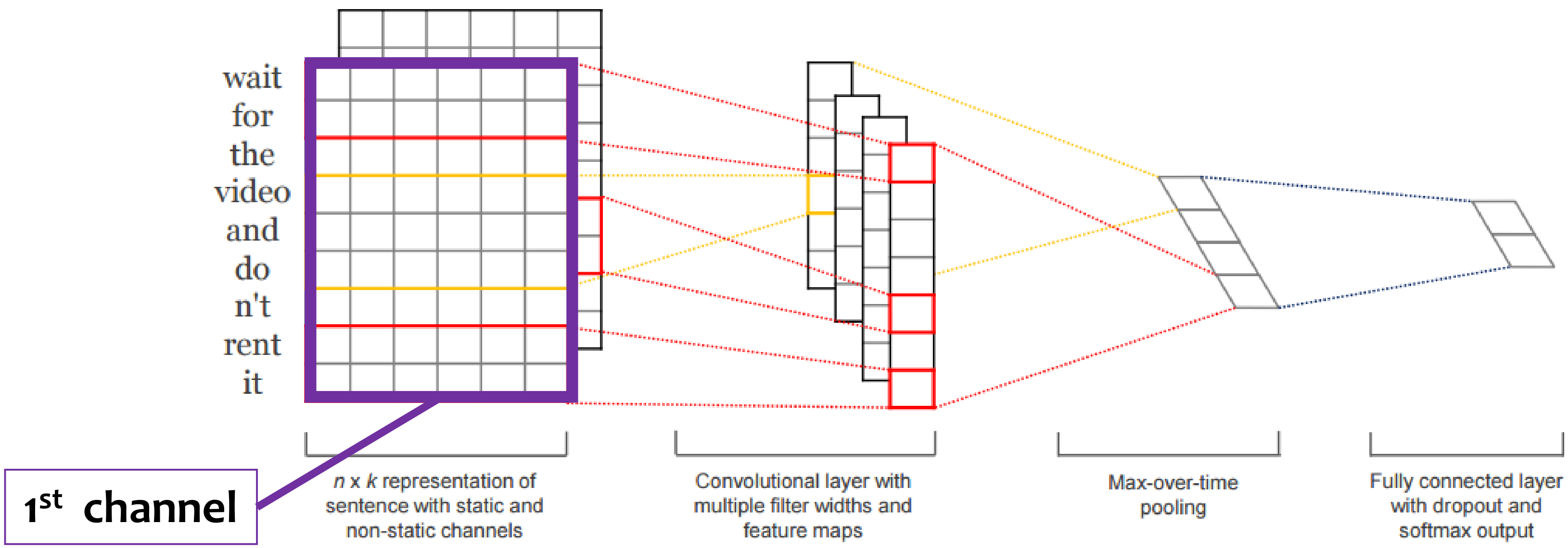


Figure 1: Model architecture with two channels for an example sentence.

- Two channels for vocabularies
  - 1st Channel: Static word vectors $v_w \in \mathbb{R}^k$
  - 2nd Channel: Fine-tuned word vectors

Sentence Modelling

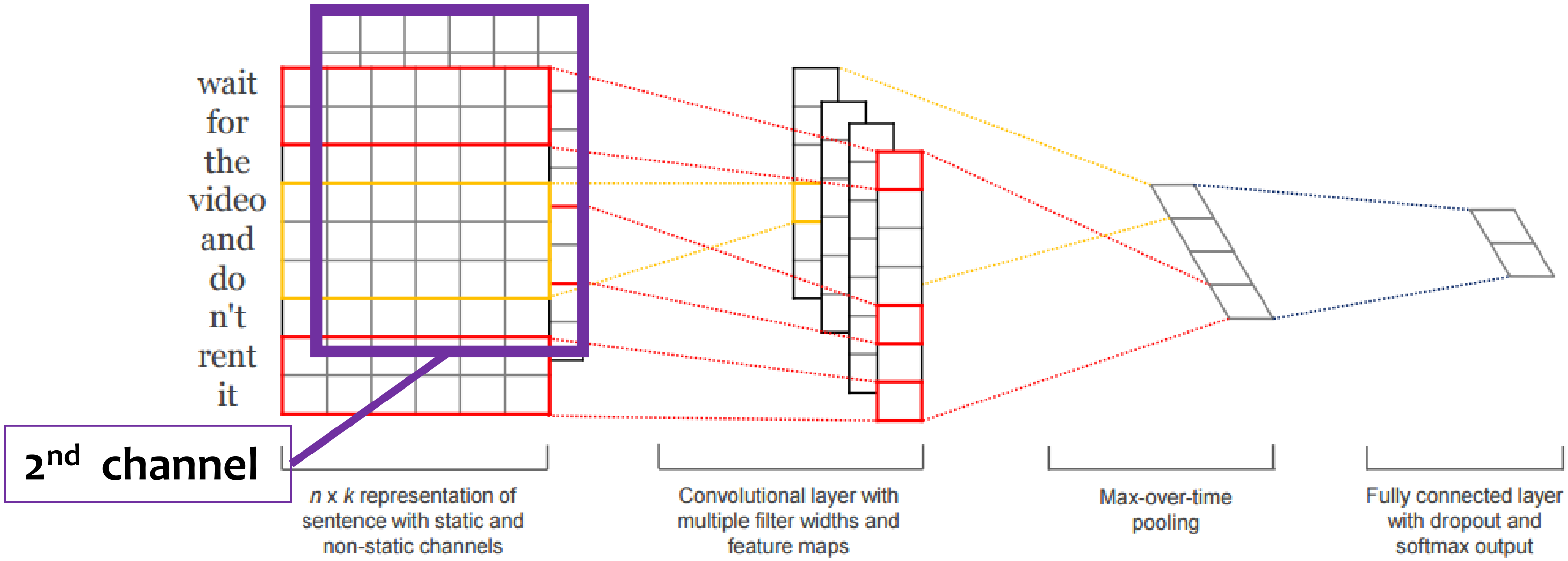


Figure 1: Model architecture with two channels for an example sentence.

- Two channels for vocabularies
  - 1st Channel: Static word vectors $v_w \in \mathbb{R}^k$
  - 2nd Channel: Fine-tuned word vectors

Sentence Modelling

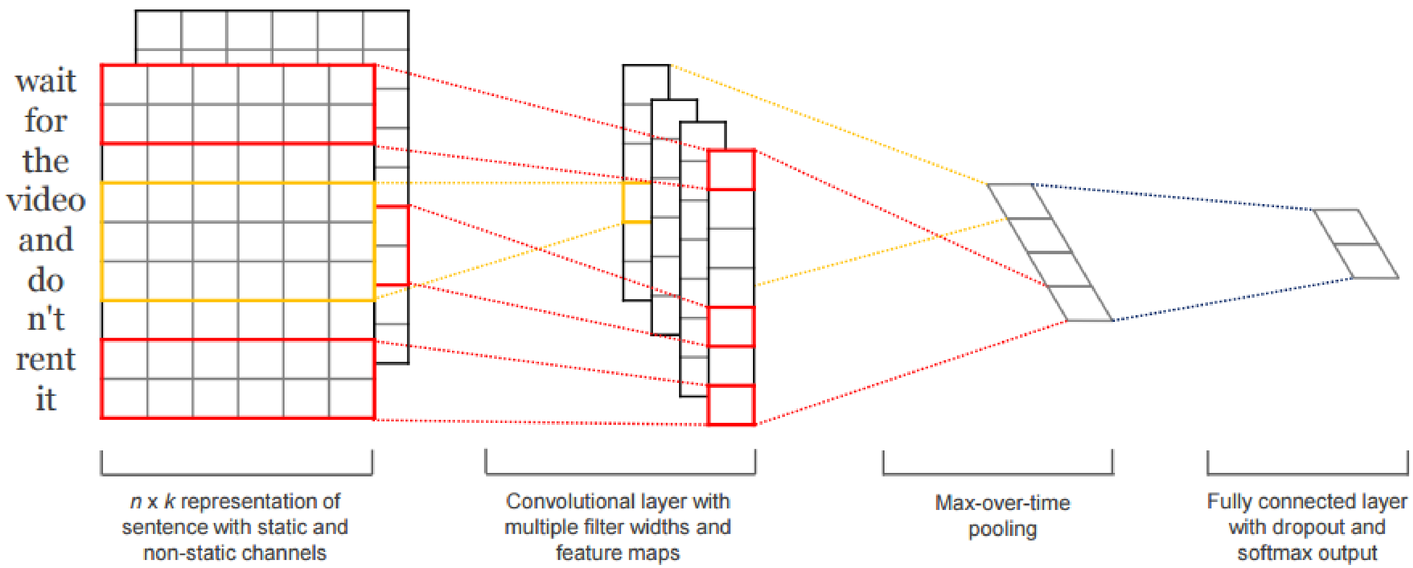


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(w x_{i:i+h-1} + b)$

$\underbrace{i:i+h-1}_{h \text{ words}}$

Sentence Modelling

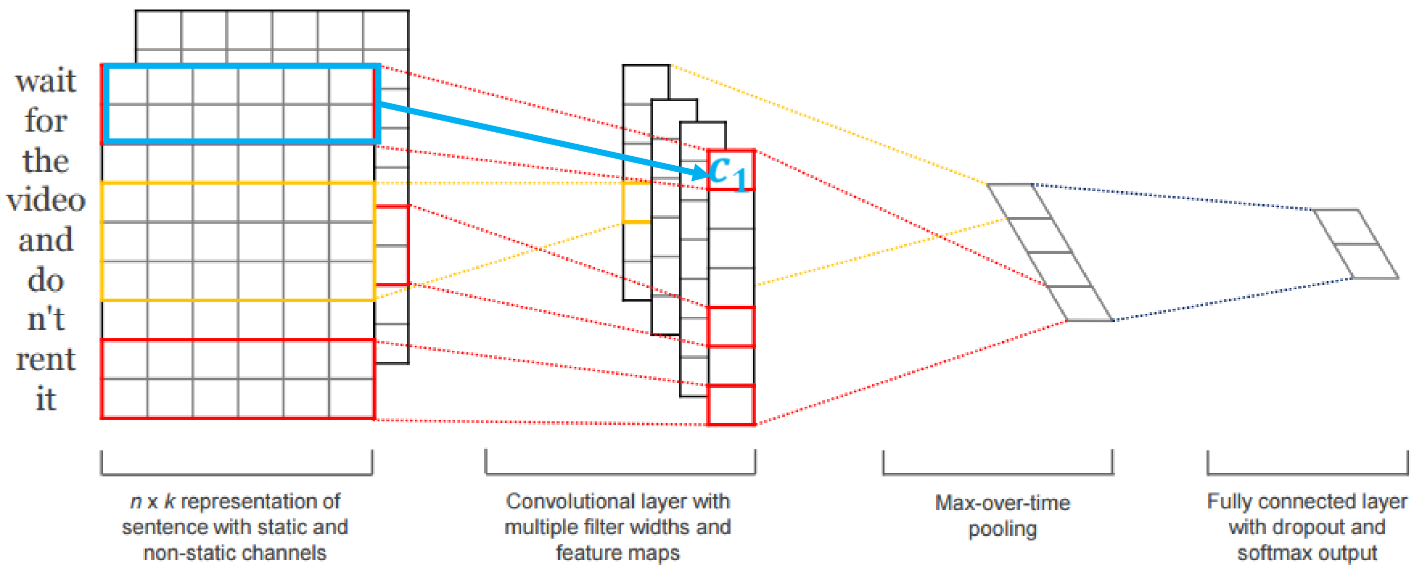


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(wx_{i:i+h-1} + b)$

$\underbrace{i:i+h-1}_{h \text{ words}}$

Sentence Modelling

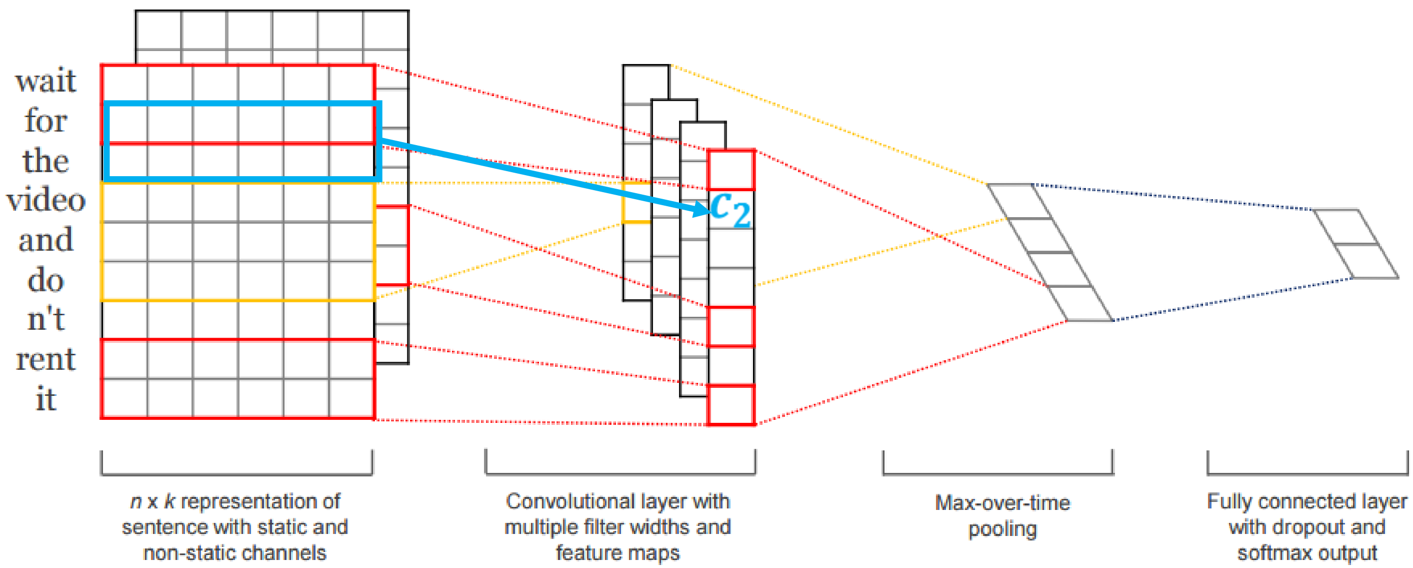


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(w x_{i:i+h-1} + b)$

h words

Sentence Modelling

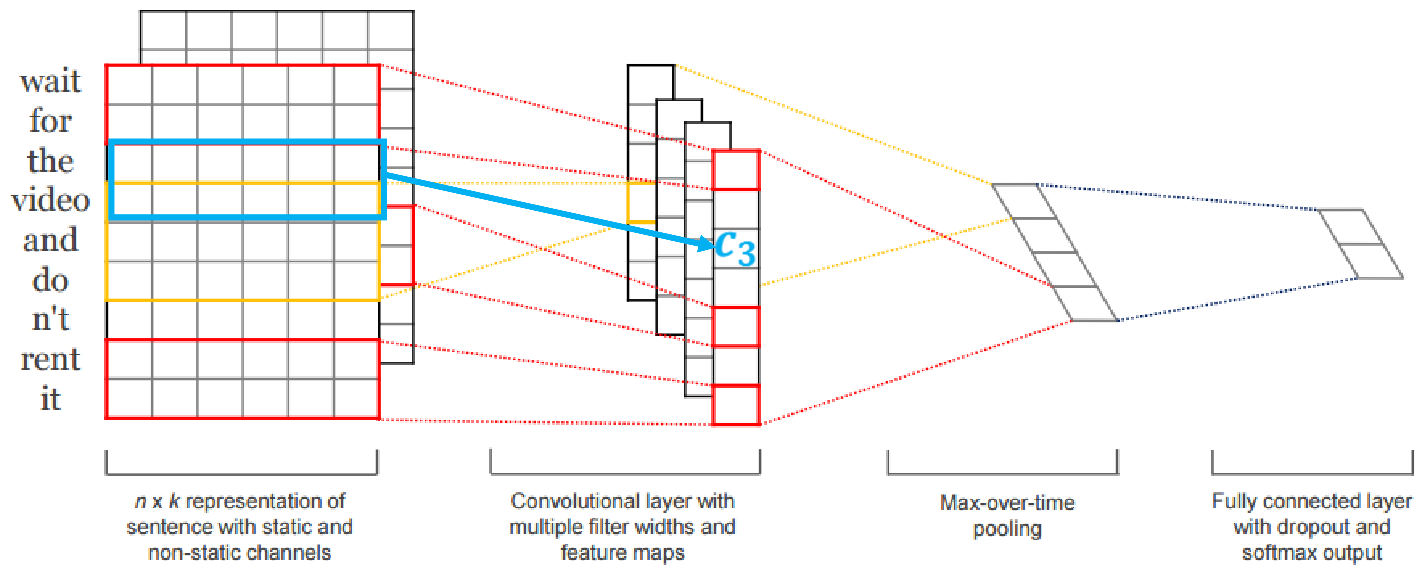


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(w x_{i:i+h-1} + b)$

    h words

Sentence Modelling

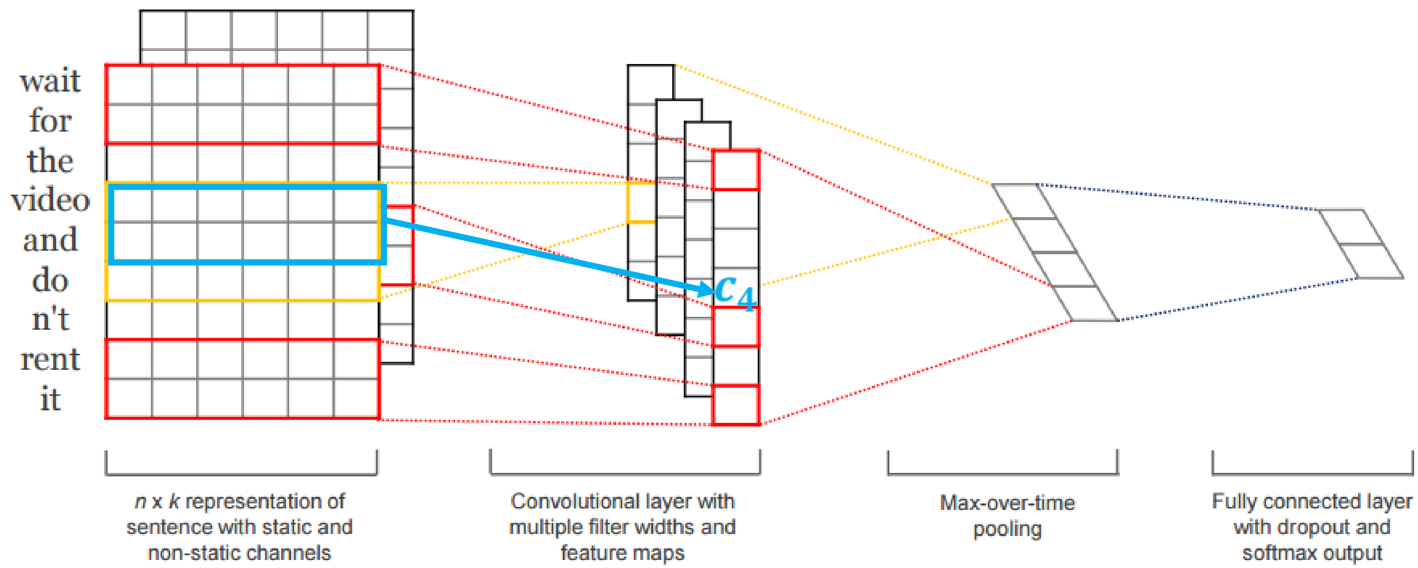


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(w \underbrace{x_{i:i+h-1}}_{h \text{ words}} + b)$

Sentence Modelling

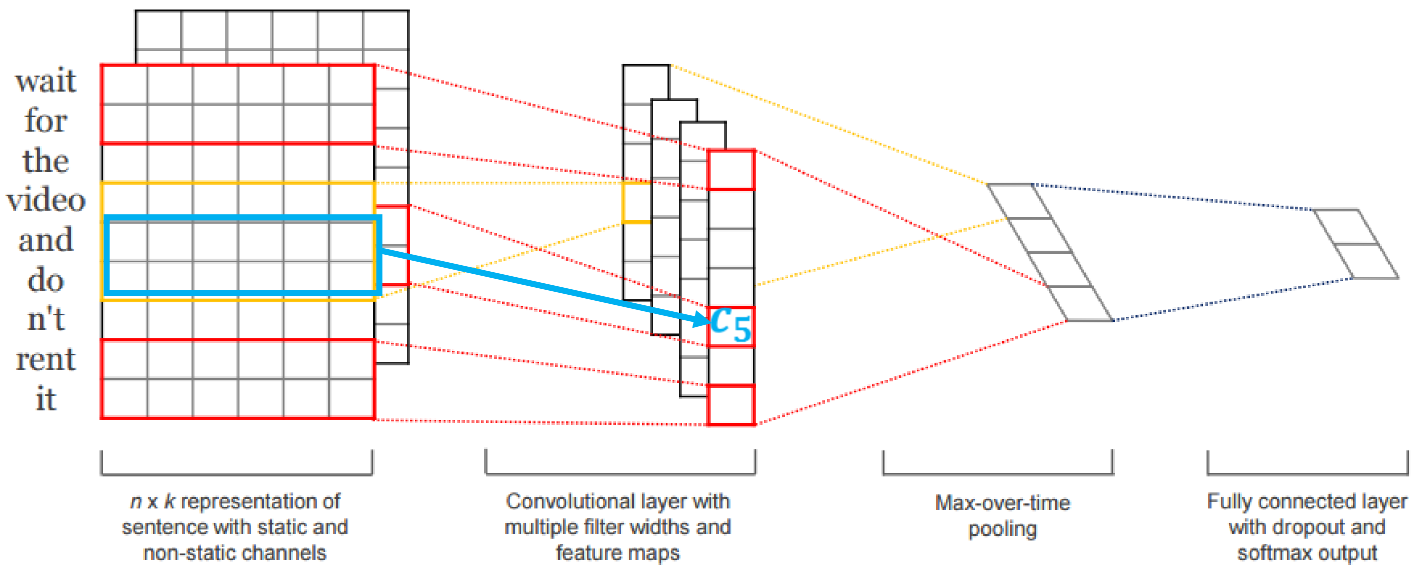


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(w x_{i:i+h-1} + b)$

h words

Sentence Modelling

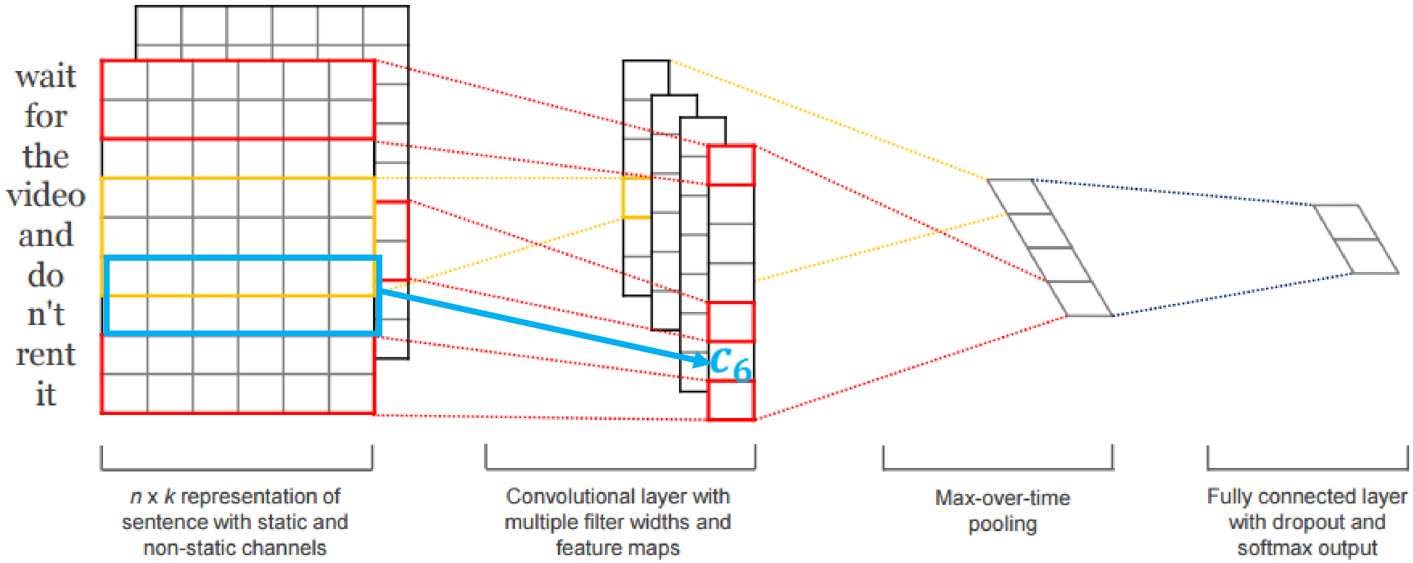


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(wx_{i:i+h-1} + b)$

h words

Sentence Modelling

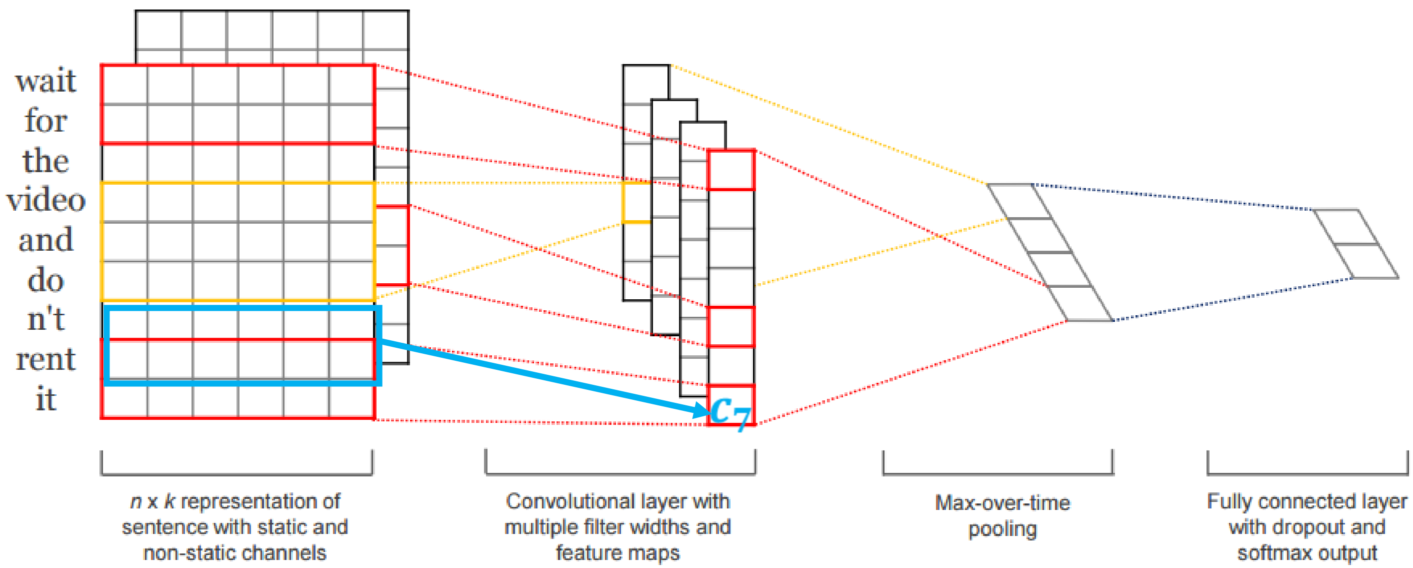


Figure 1: Model architecture with two channels for an example sentence.

- One filter $w \in \mathbb{R}^{hk}$ for h words Convolution
  - $c_i = f(w x_{i:i+h-1} + b)$

h words

Sentence Modelling

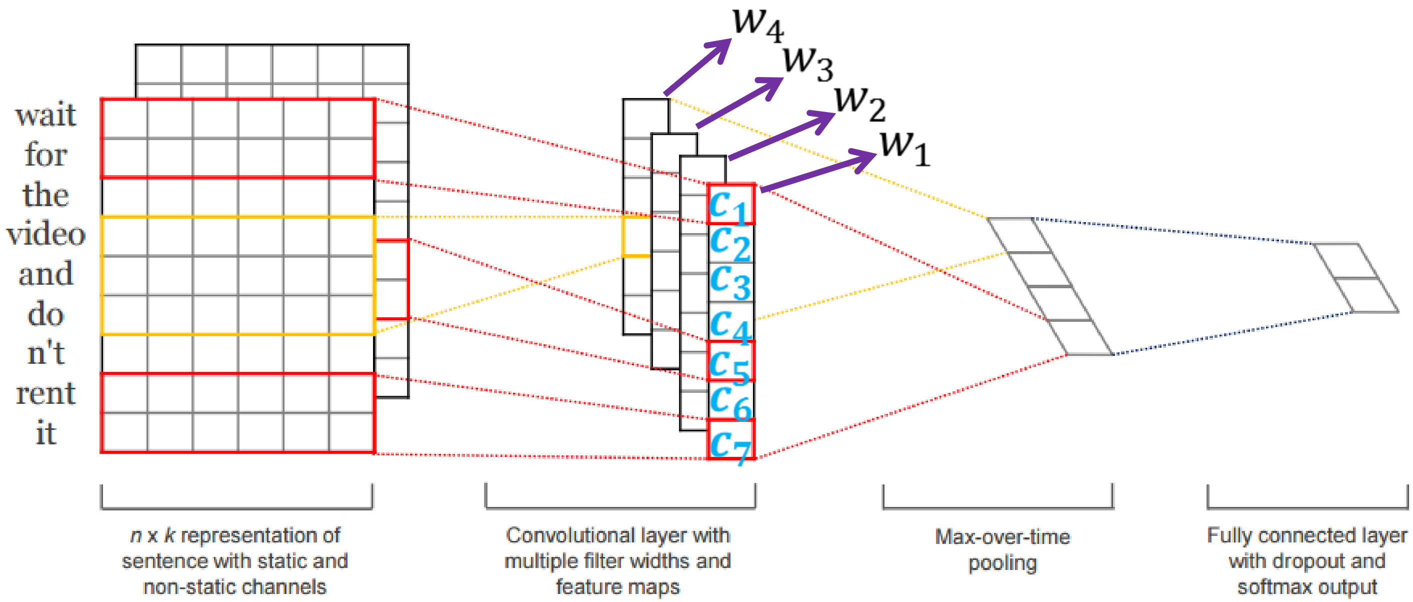


Figure 1: Model architecture with two channels for an example sentence.

- Feature map $c = (c_1, c_2, c_3, c_4, c_5, c_6, c_7)$
- Different filters $w_1, \dots w_4$, produce different feature maps, so the length will vary

Sentence Modelling

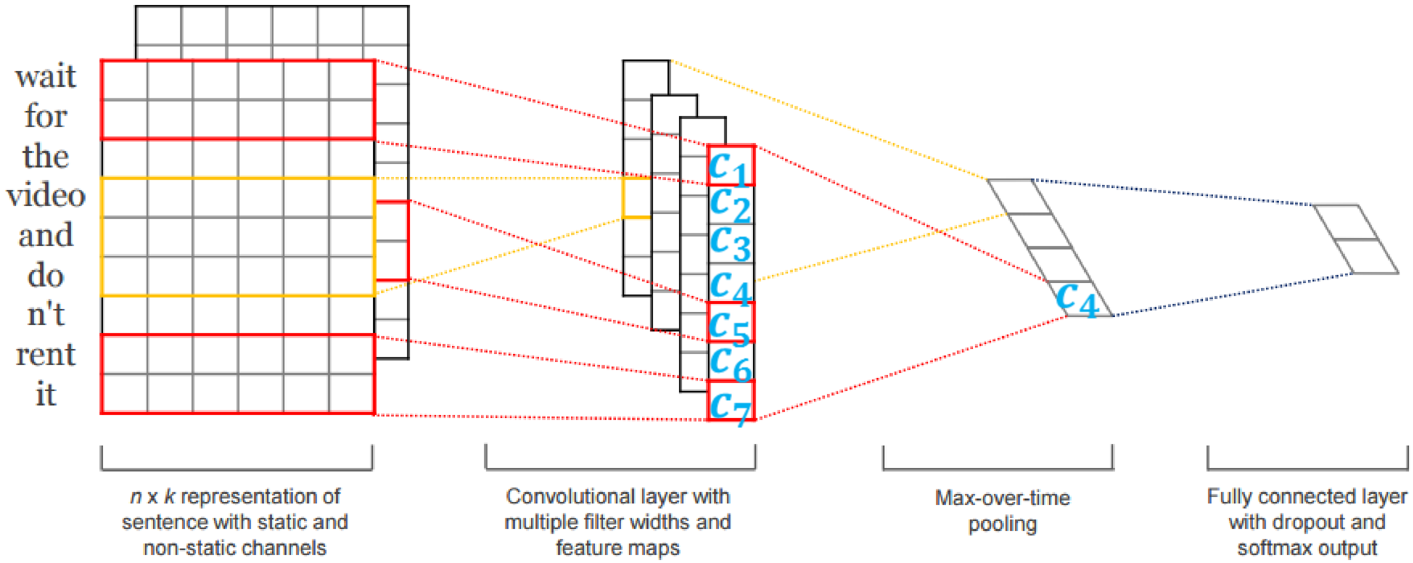


Figure 1: Model architecture with two channels for an example sentence.

- Max pooling $\hat{c} = \max\{c\}$
- Four filters end up with a four dim feature vector, suppose c_4 is largest

Sentence Modelling

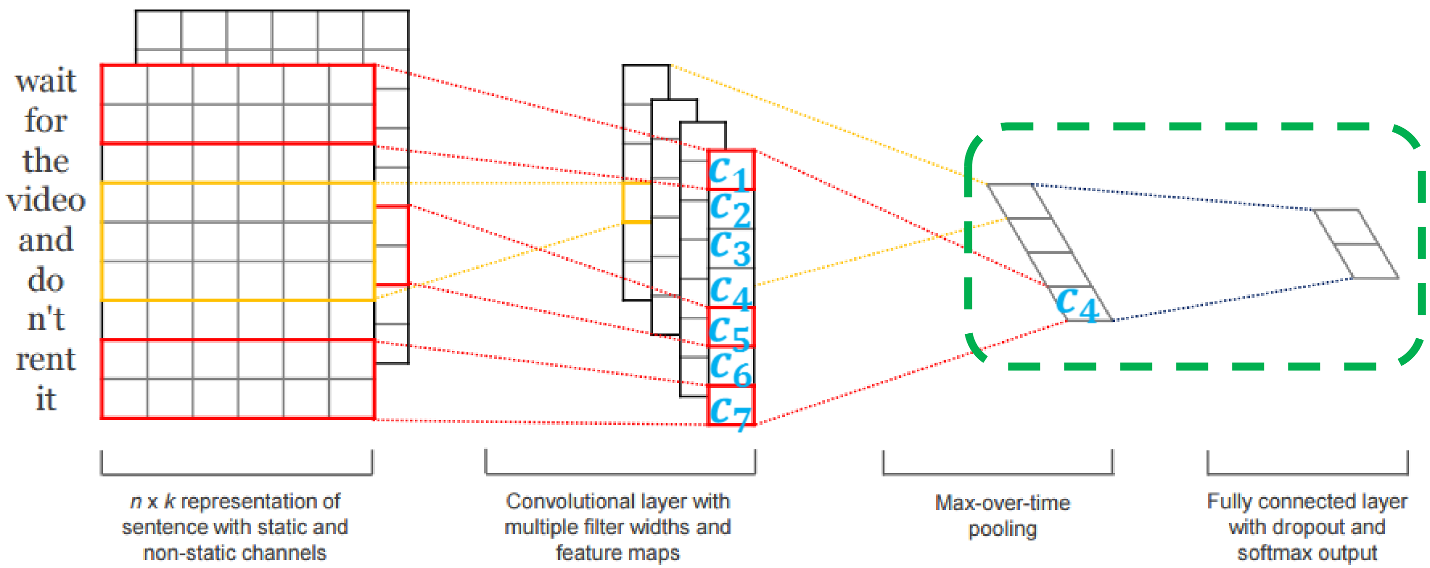


Figure 1: Model architecture with two channels for an example sentence.

- Fully connected layers for final classification
  - SoftMax for multiclass classification
  - Logistic sigmoid for binary classification

Sentence Modelling

- Experiment
  - “For all datasets we use: Rectified linear units, filter windows (h) of 3,4,5 with 100 feature maps each, dropout rate (p) of 0.5, l_2 constraints (s) of 3, […]”

Sentence Modelling

- Sentiment classification

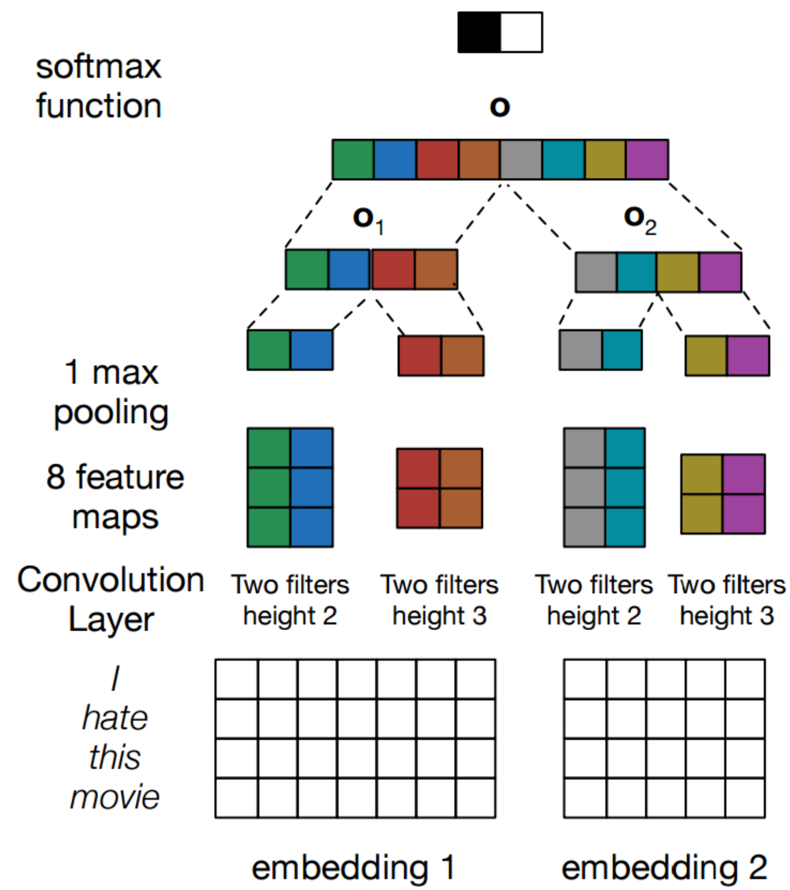
- Question classification

Phrasal Modelling

- **MGNC-CNN: a simple approach to exploiting multiple word embeddings for sentence classification, ACL 2016**
  - Different sources of word embeddings

Phrasal Modelling

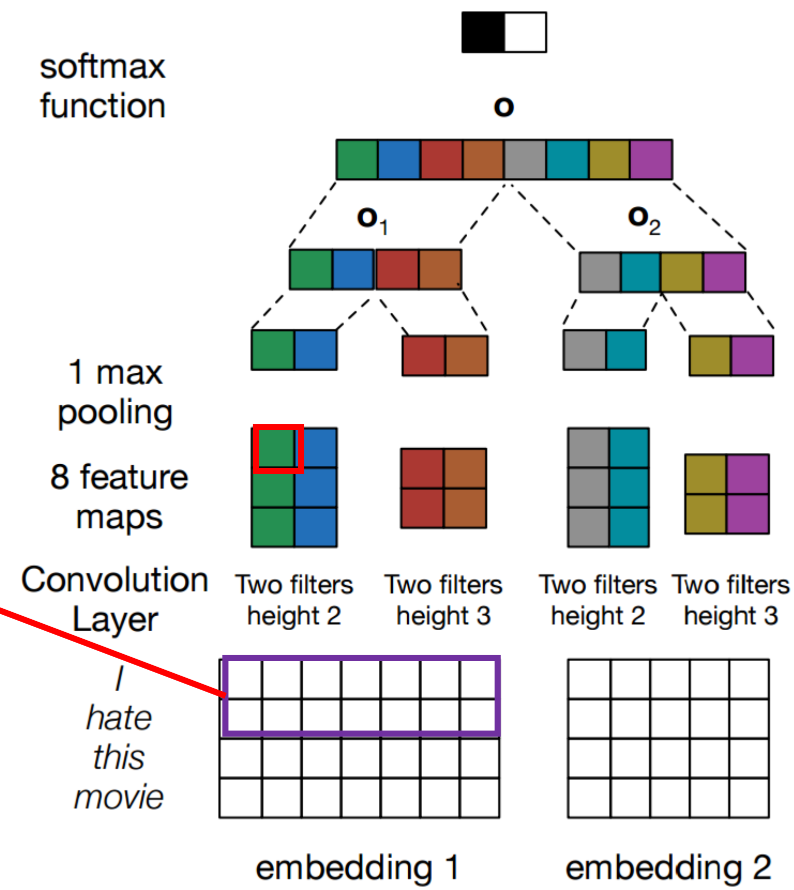
- Motivation
  - "many pretrained word embeddings are now readily available on the web, induced using different models, corpora, and processing steps."
  - "Different embeddings may encode different aspects of language : those based on BoW statistics tend to capture associations (*doctor* and *hospital*) while embeddings based on dependency-parses encode similarity in terms of use (*doctor* and *surgeon*)."
  - "It is natural to consider how these embeddings might be *combined* to improve NLP models in general and CNNs in particular."



Phrasal Modelling

- Filter

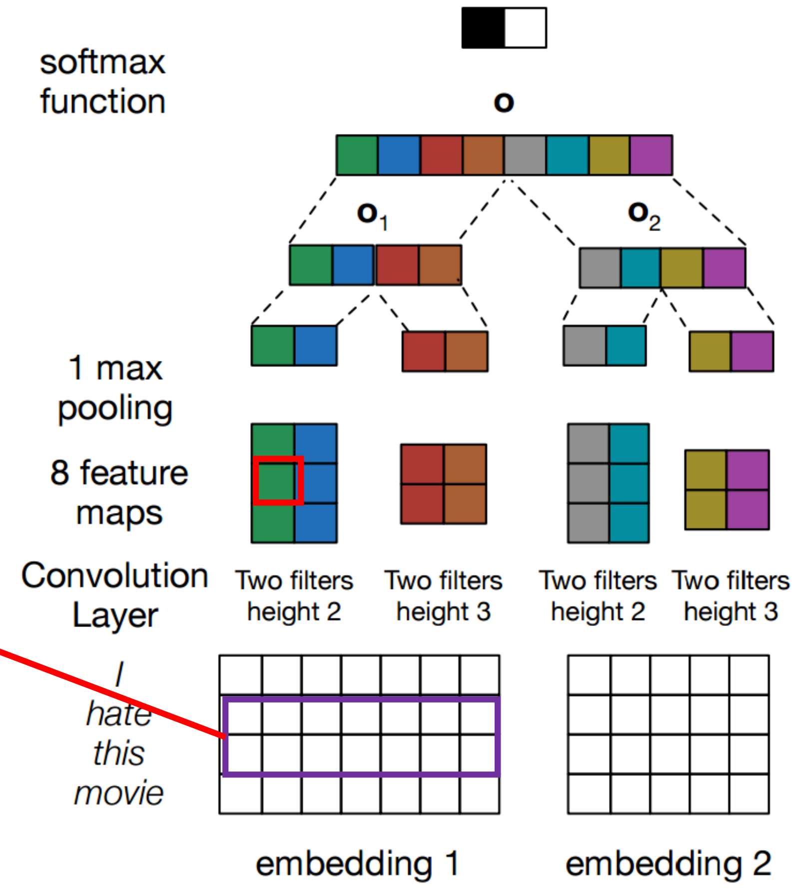
A filter with height 2



Phrasal Modelling

- Filter

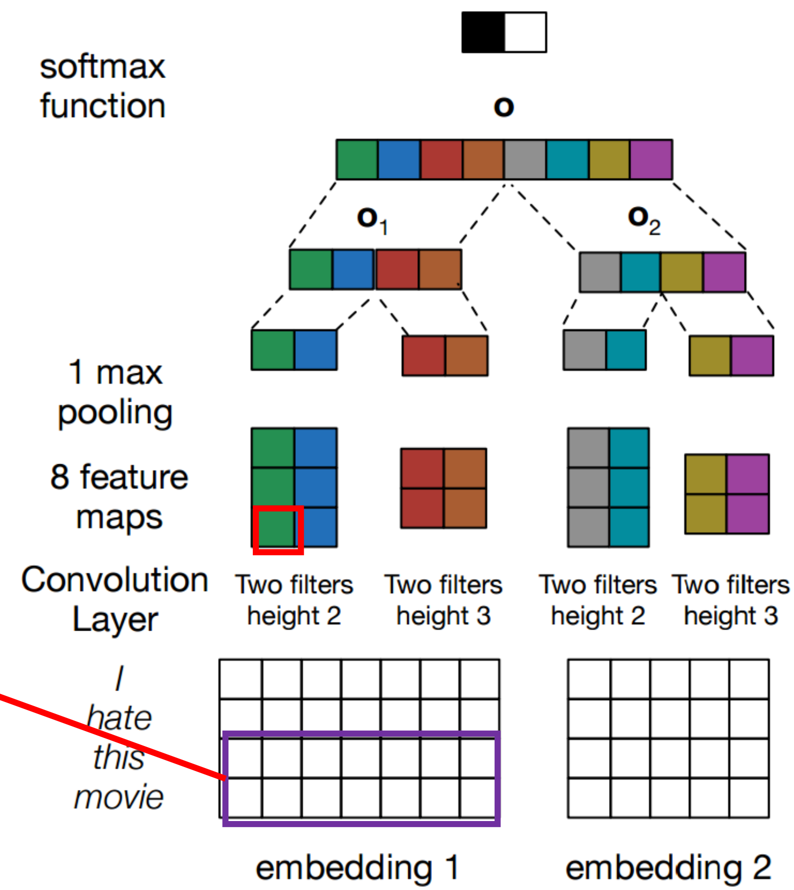
A filter with height 2



Phrasal Modelling

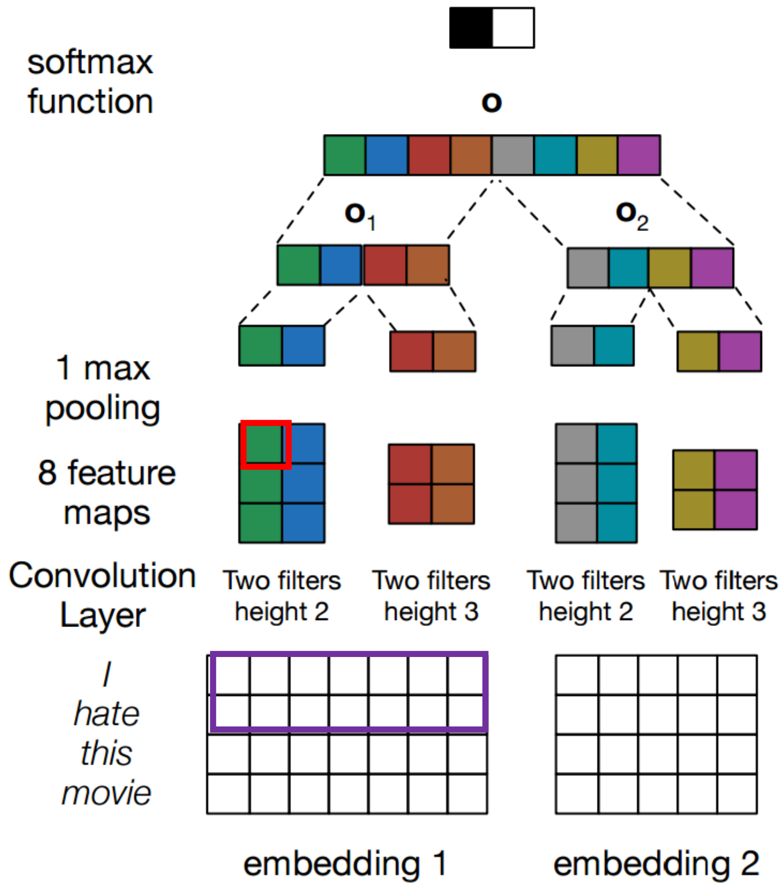
- Filter

A filter with height 2



Phrasal Modelling

- Filter groups



Sentence Modelling

- **A convolutional neural network for modelling sentences, ACL 2014**
  - Sentiment prediction
    - movie reviews
    - Twitter with distant supervision
  - Question type classification
- Dynamic k-max pooling
  - As a way of feature selection

Sentence Modelling

- “The aim of a sentence model is to analyze and represent the semantic content of a sentence for purposes of classification and generation.“
- “one must represent a sentence in terms of features that depend on the words and short n-grams that are frequently observed. The core of a sentence model involves a feature function that defines the process […]”

Sentence Modelling

- Composition of word-level feature vectors is one way leading to represent phrasal-sentential-level features
- Then, what is compositionality?

Sentence Modelling

- Composition of word-level feature vectors is one way leading to represent phrasal-sentential-level features
- Then, what is compositionality?

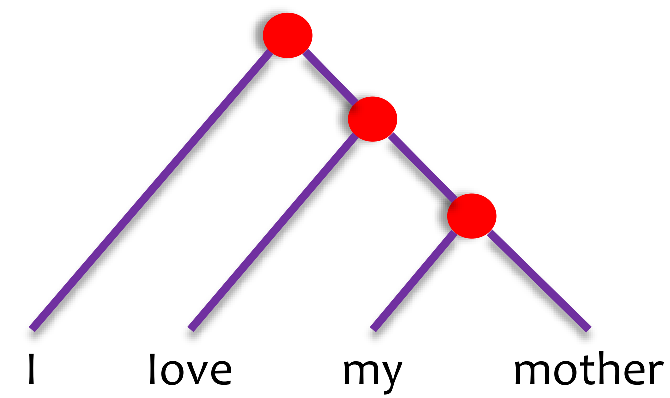
Principle of compositionality

meaning of a complex expression is determined by the meanings of its constituent expressions and the rules used to combine them.

Originated from Gotlob Frege

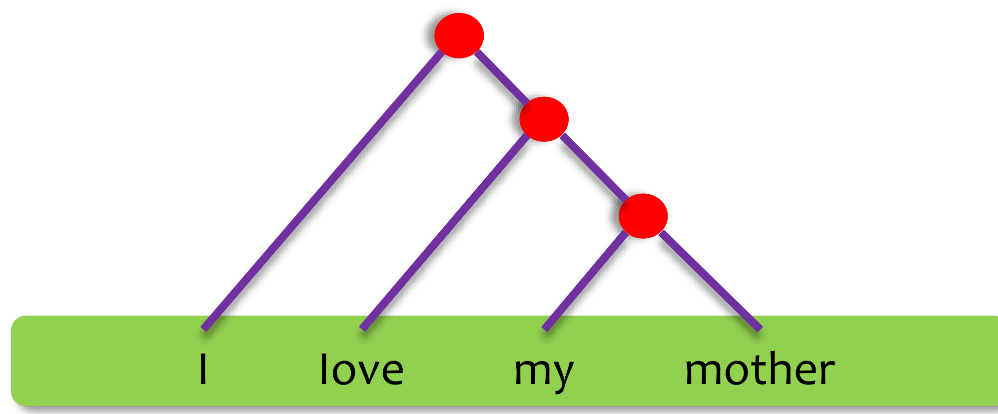
Sentence Modelling

- Think about a very simple way of composing!
  - Motivated from formal semantics, Montague



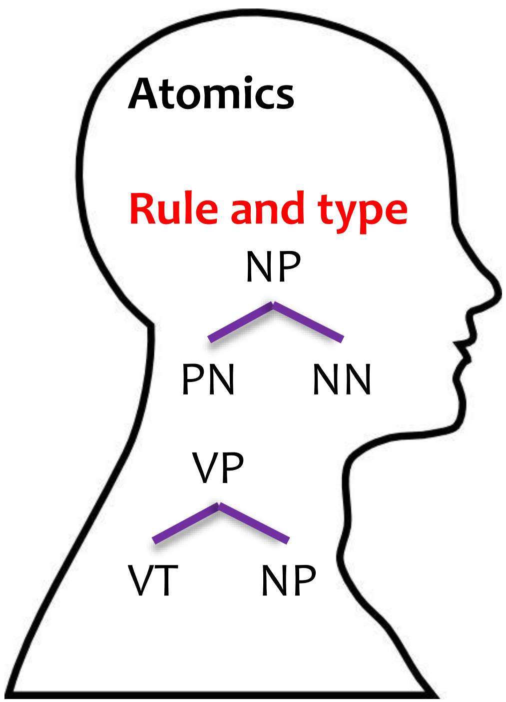
Sentence Modelling

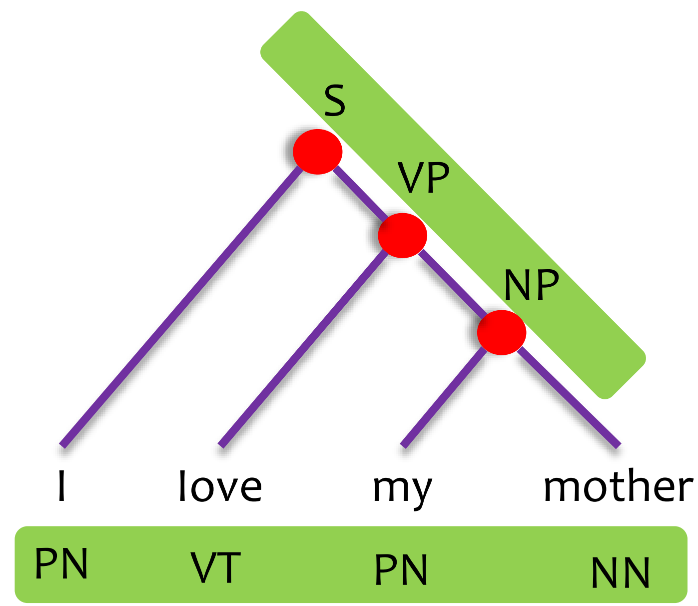
- Think about a very simple way of composing!
  - Motivated from formal semantics, Montague



Sentence Modelling

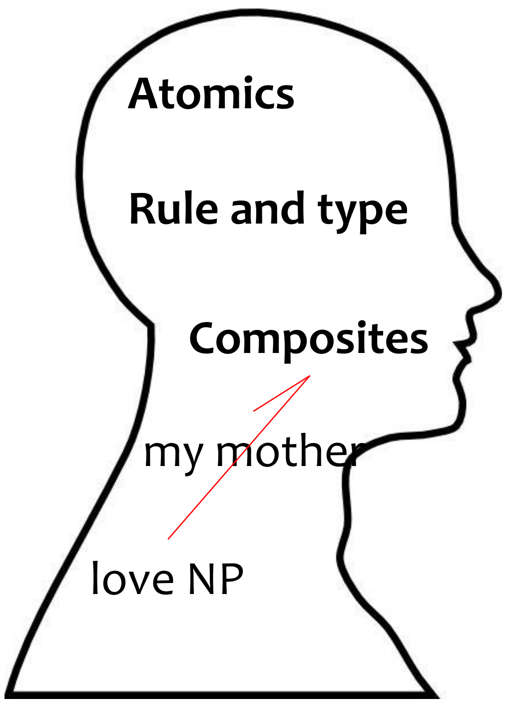
- Think about a very simple way of composing!
  - Motivated from formal semantics, Montague

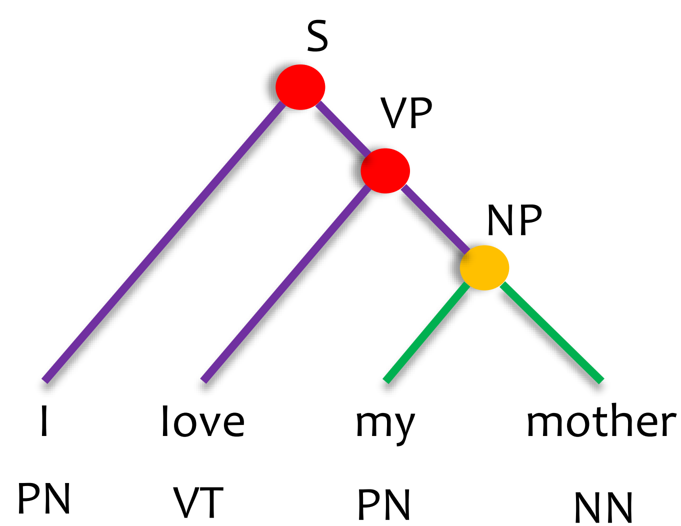




Sentence Modelling

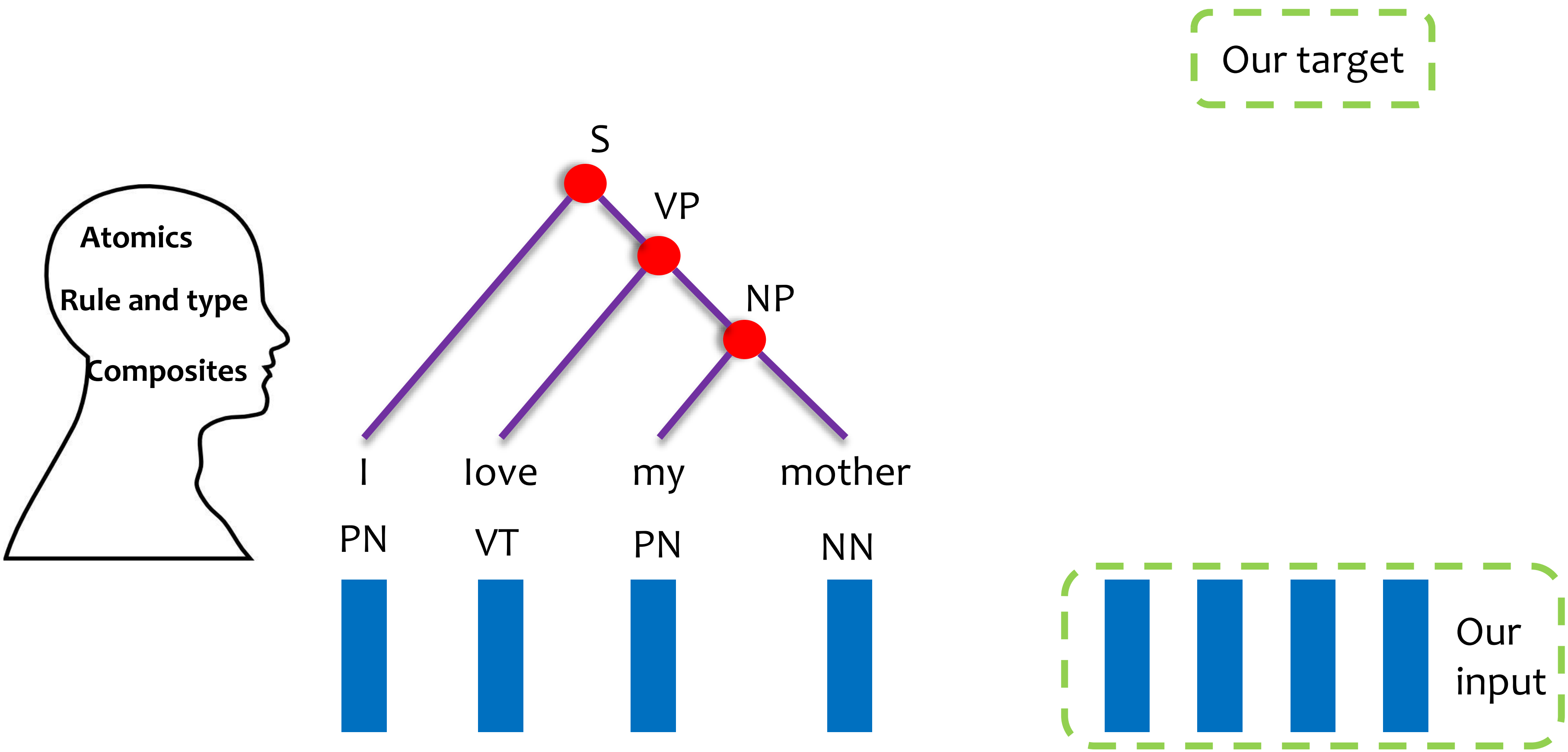
- Think about a very simple way of composing!
  - Motivated from formal semantics, Montague





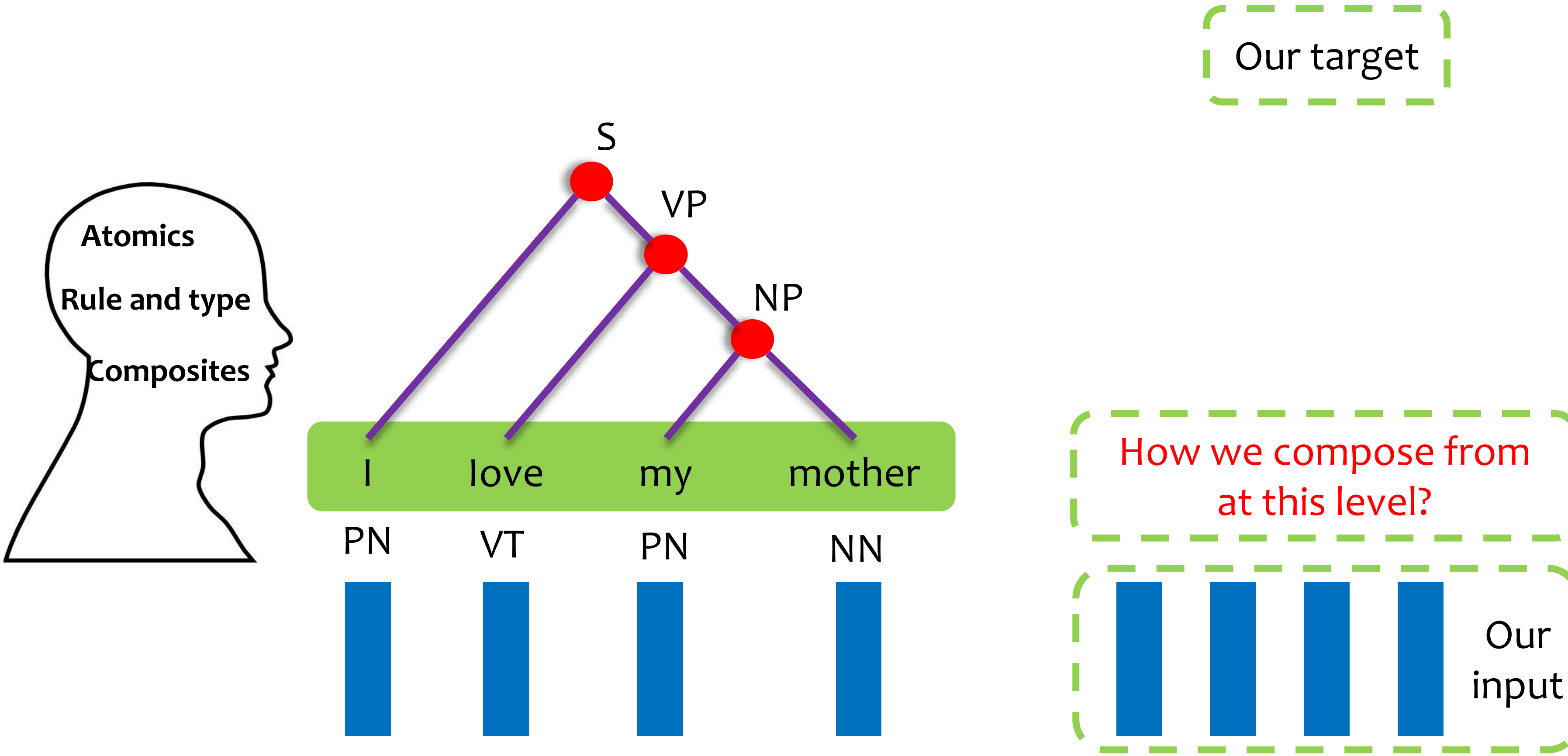
Sentence Modelling

- We have feature vectors for word meaning
  - Word2vec, GloVe, etc.



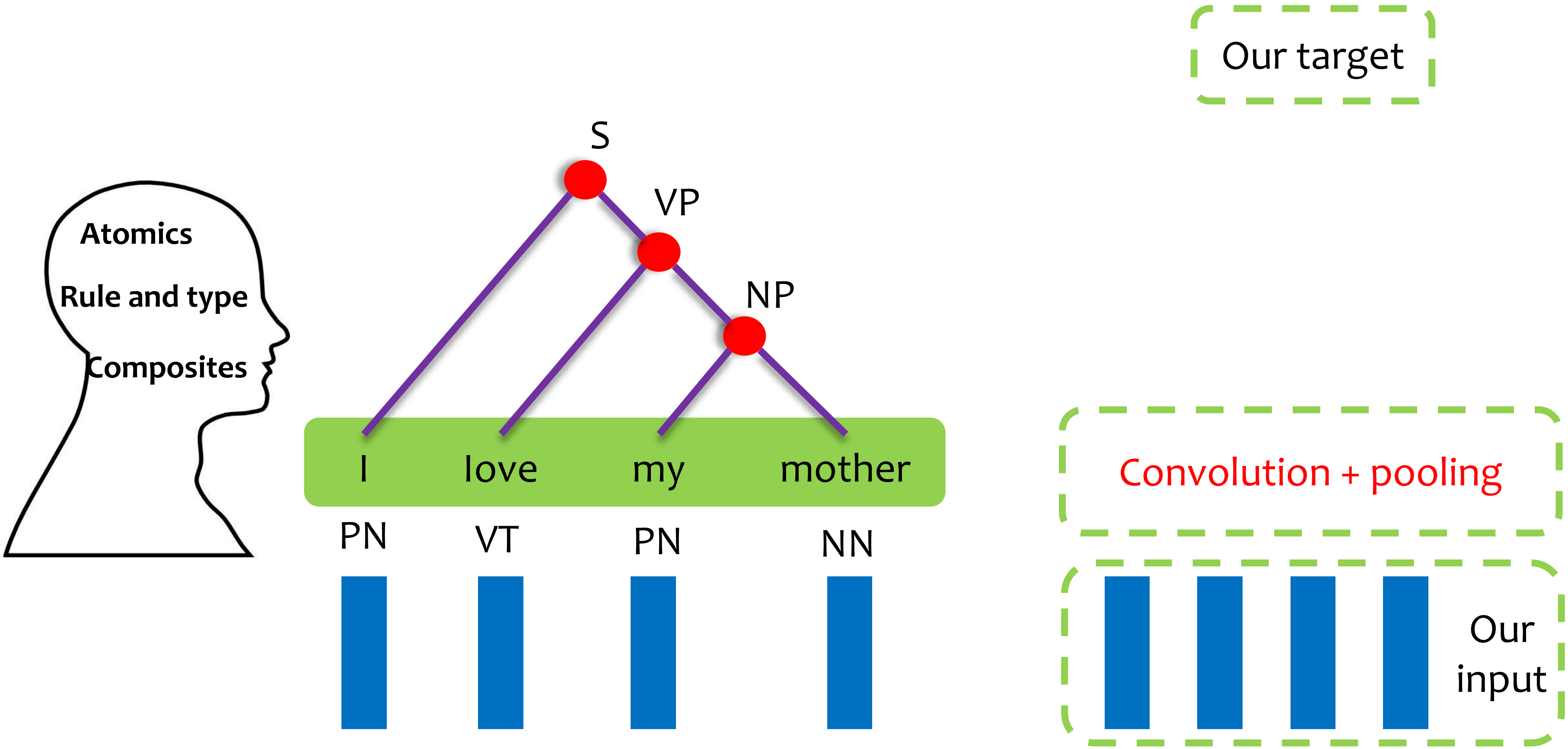
Sentence Modelling

- We have feature vectors for word meaning
  - Word2vec, GloVe, etc.



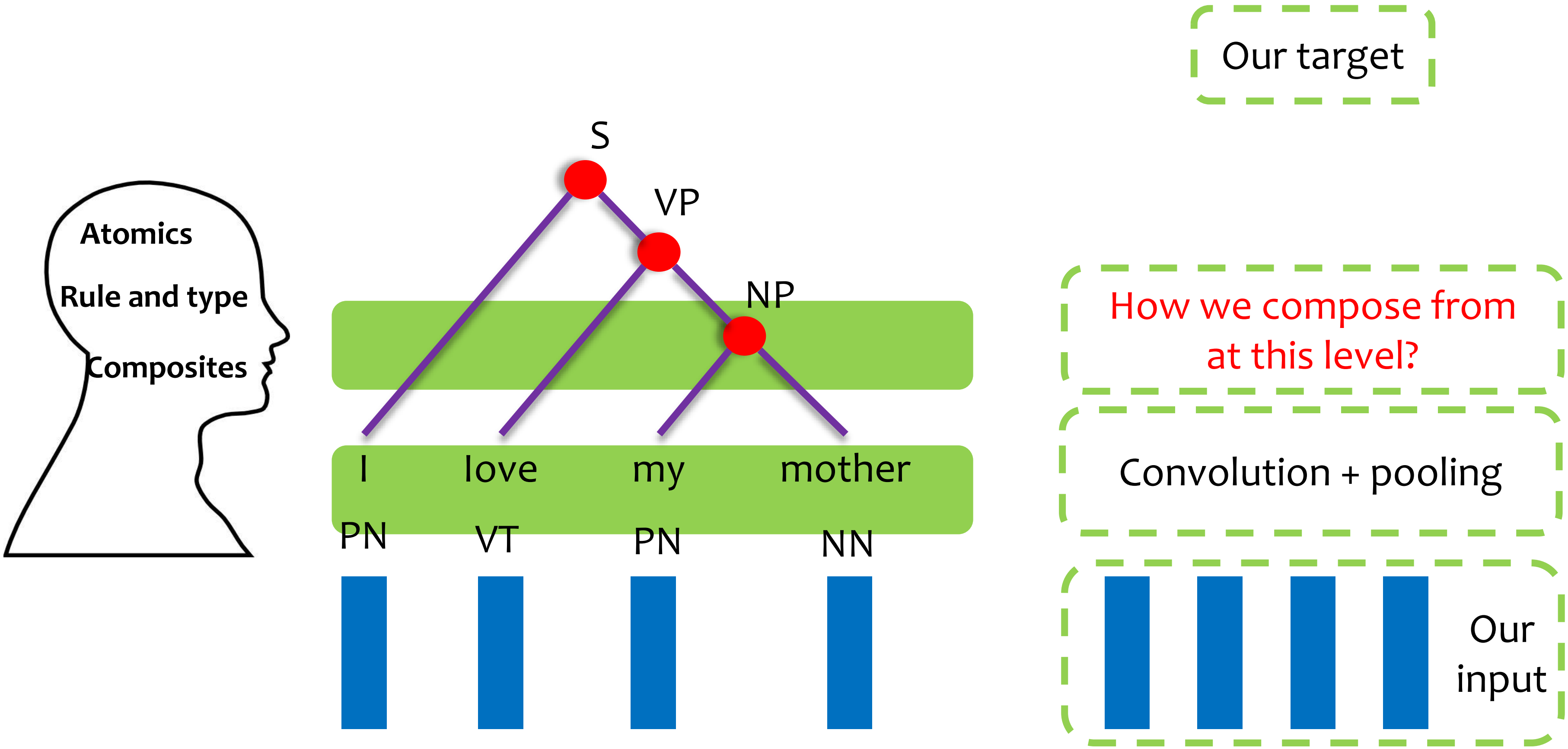
Sentence Modelling

- We have feature vectors for word meaning
  - Word2vec, GloVe, etc.



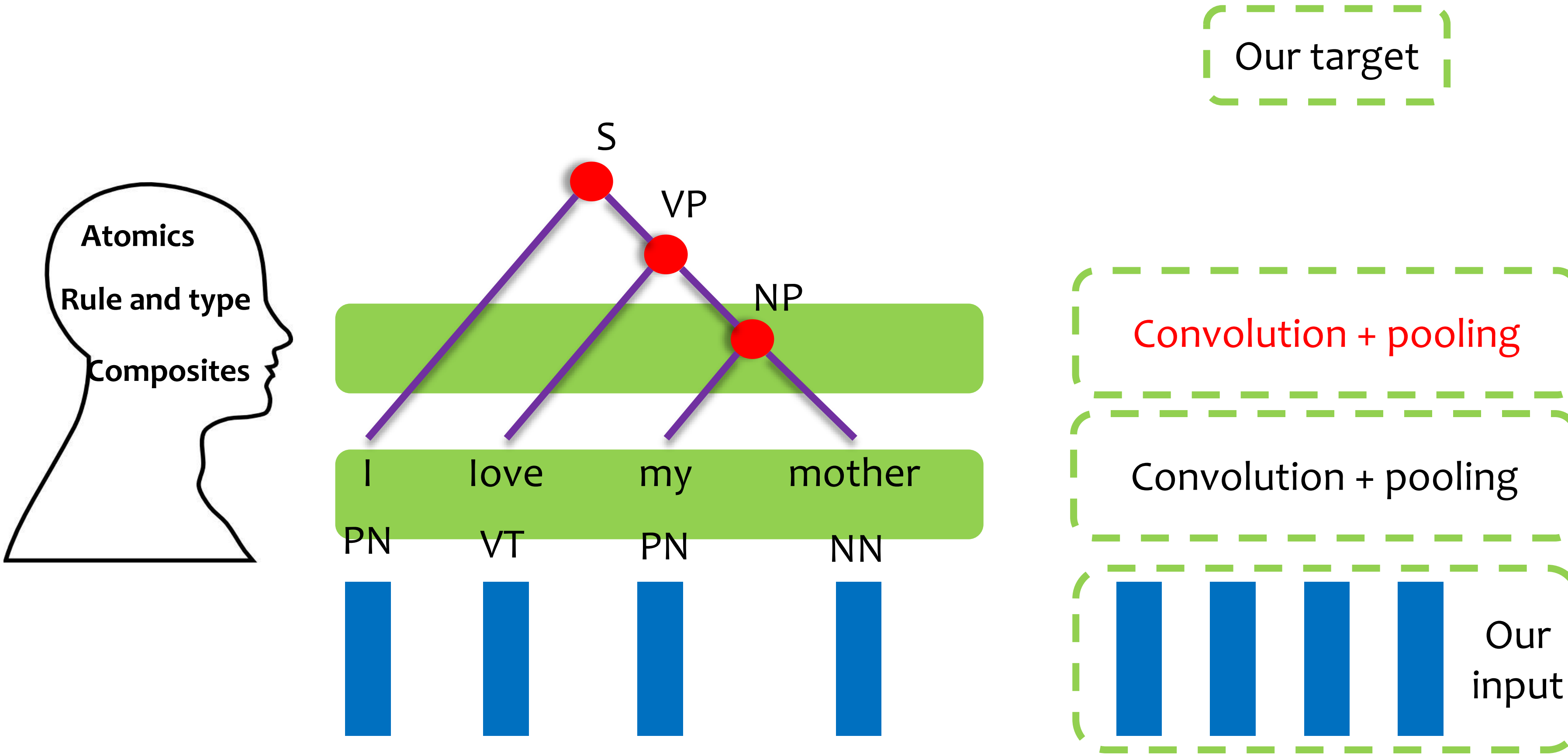
Sentence Modelling

- We have feature vectors for word meaning
  - Word2vec, GloVe, etc.



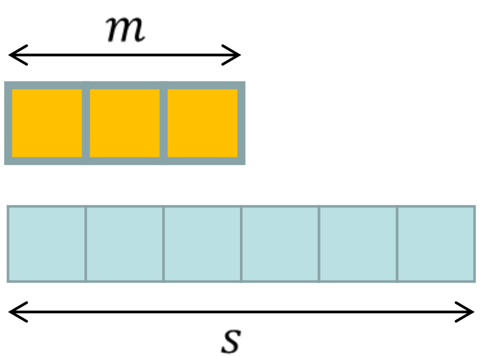
Sentence Modelling

- We have feature vectors for word meaning
  - Word2vec, GloVe, etc.



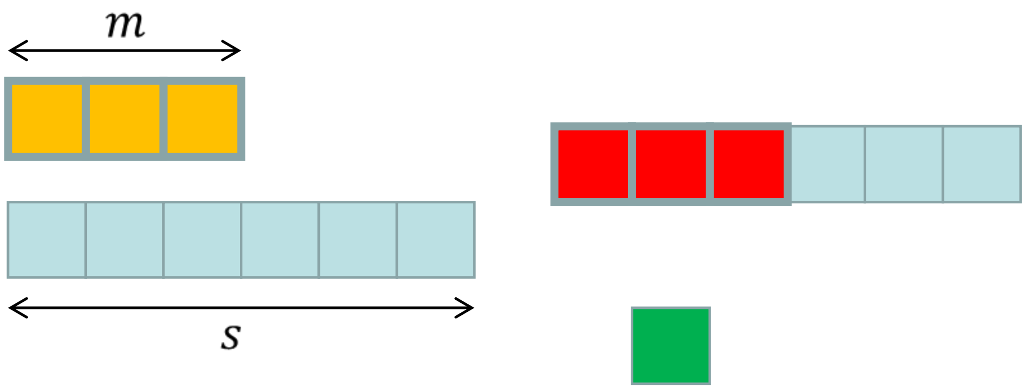
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow s - m + 1$, where $s > m$



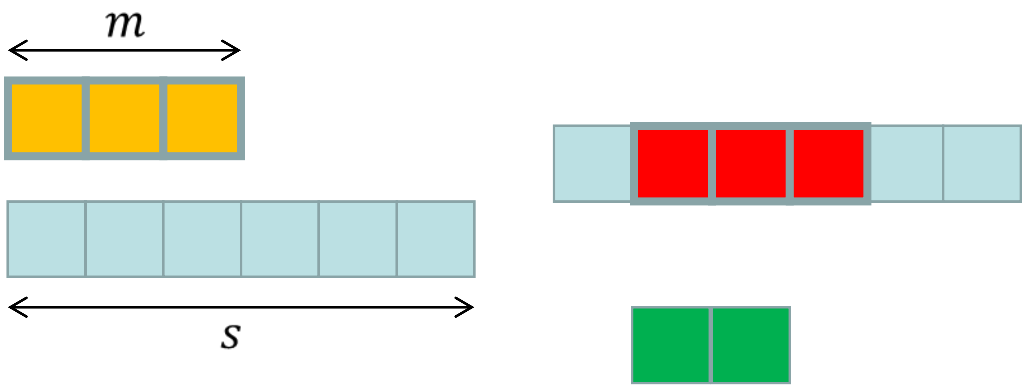
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow s - m + 1$, where $s > m$



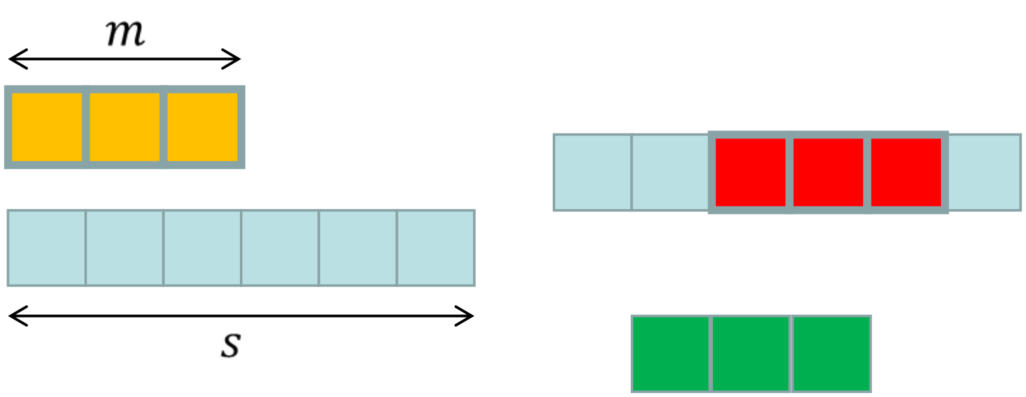
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow s - m + 1$, where $s > m$



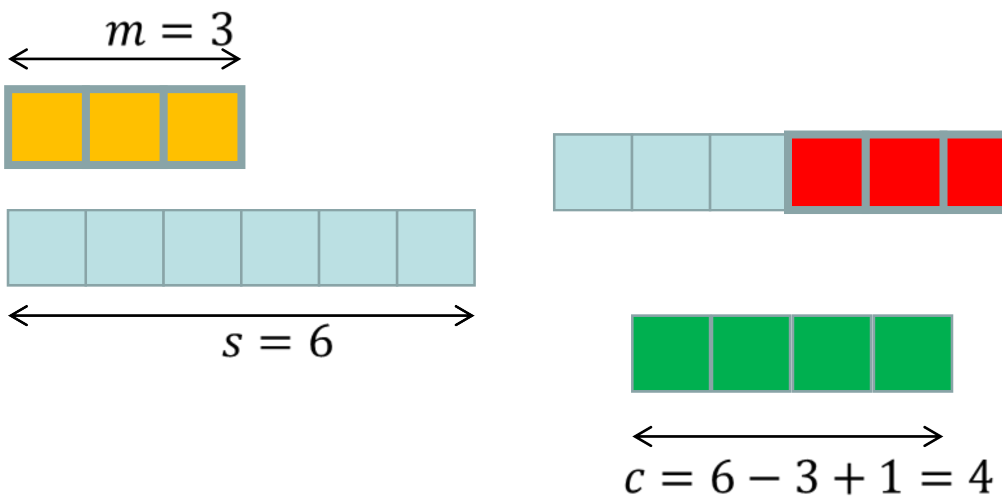
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow s - m + 1$, where $s > m$



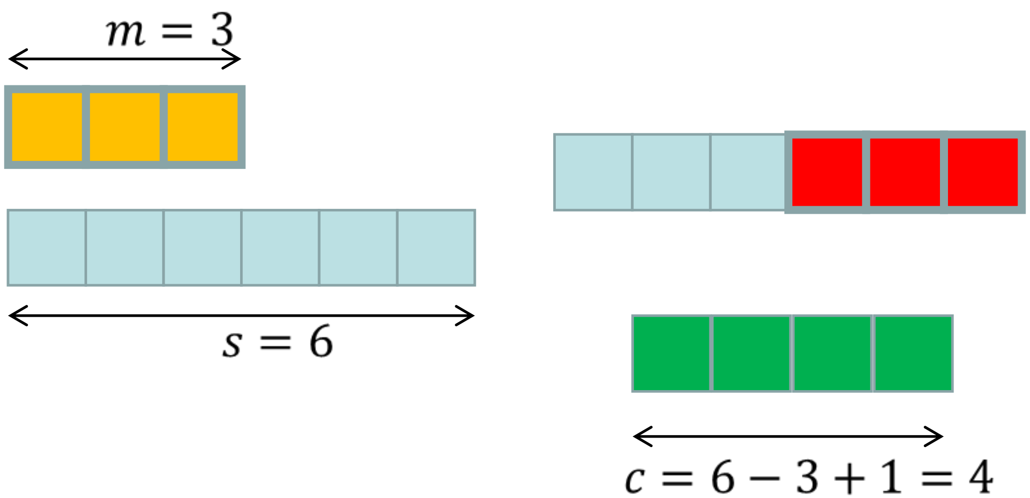
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$



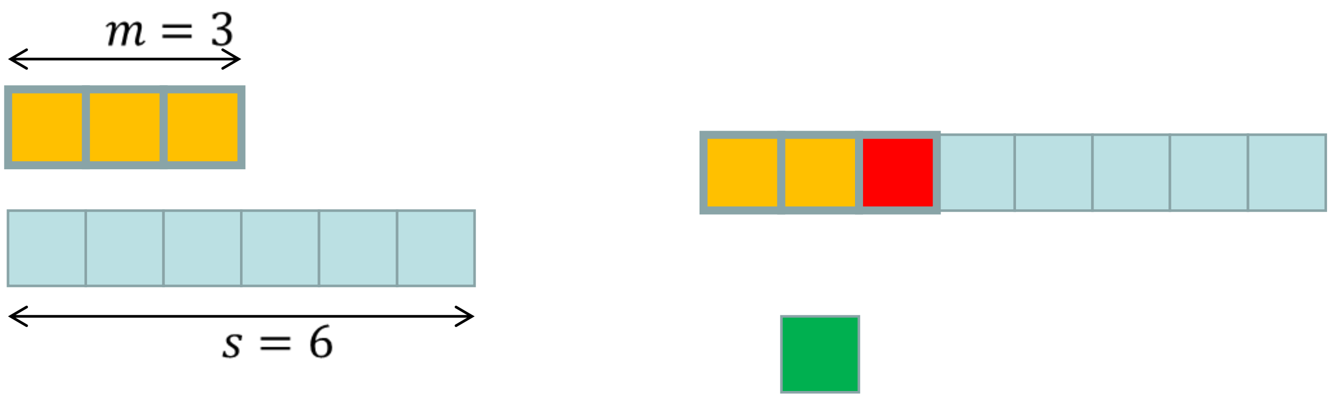
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$



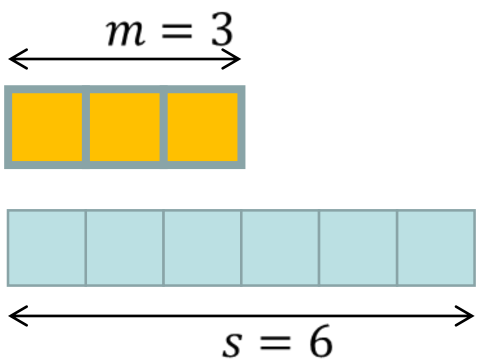
Sentence Modelling

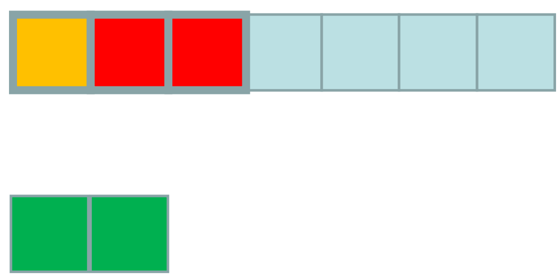
- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$



Sentence Modelling

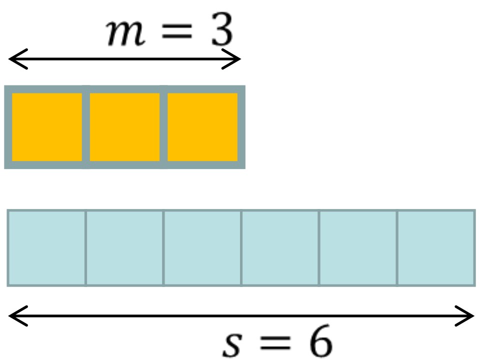
- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$

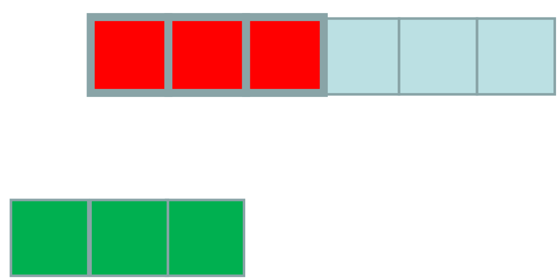




Sentence Modelling

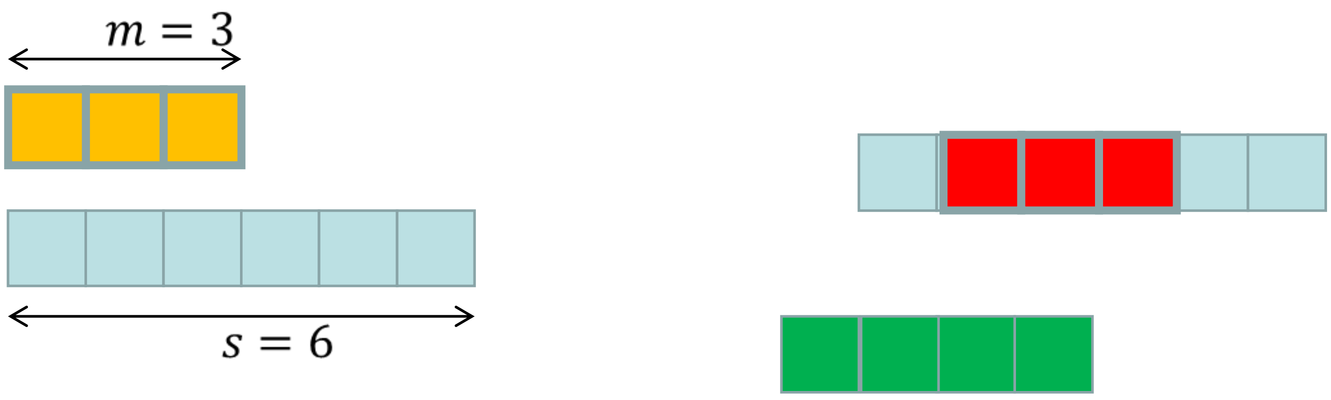
- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$





Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$

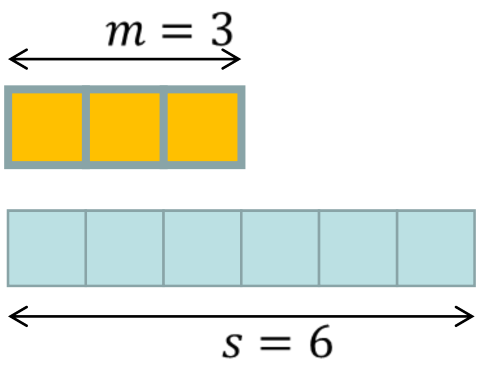


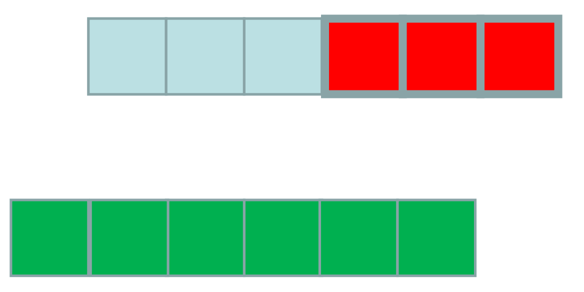
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$

Sentence Modelling

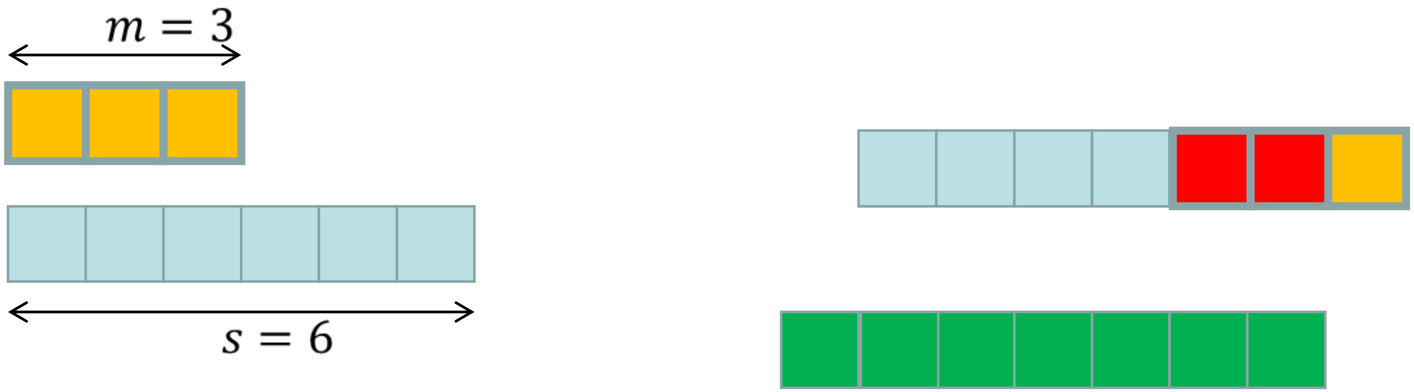
- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$





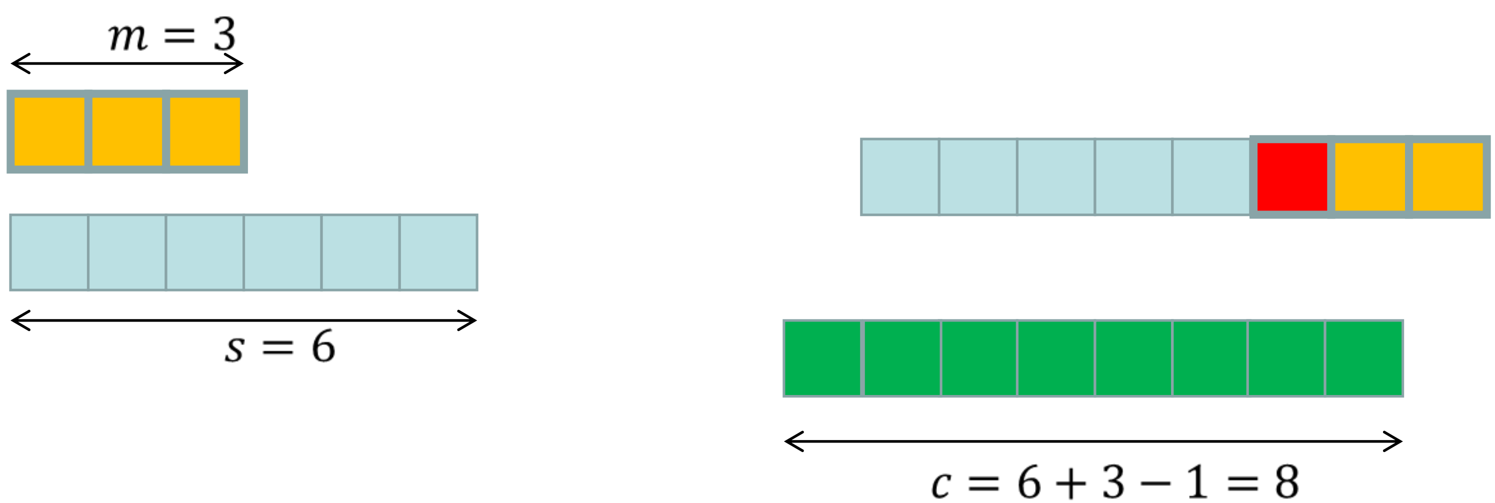
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$



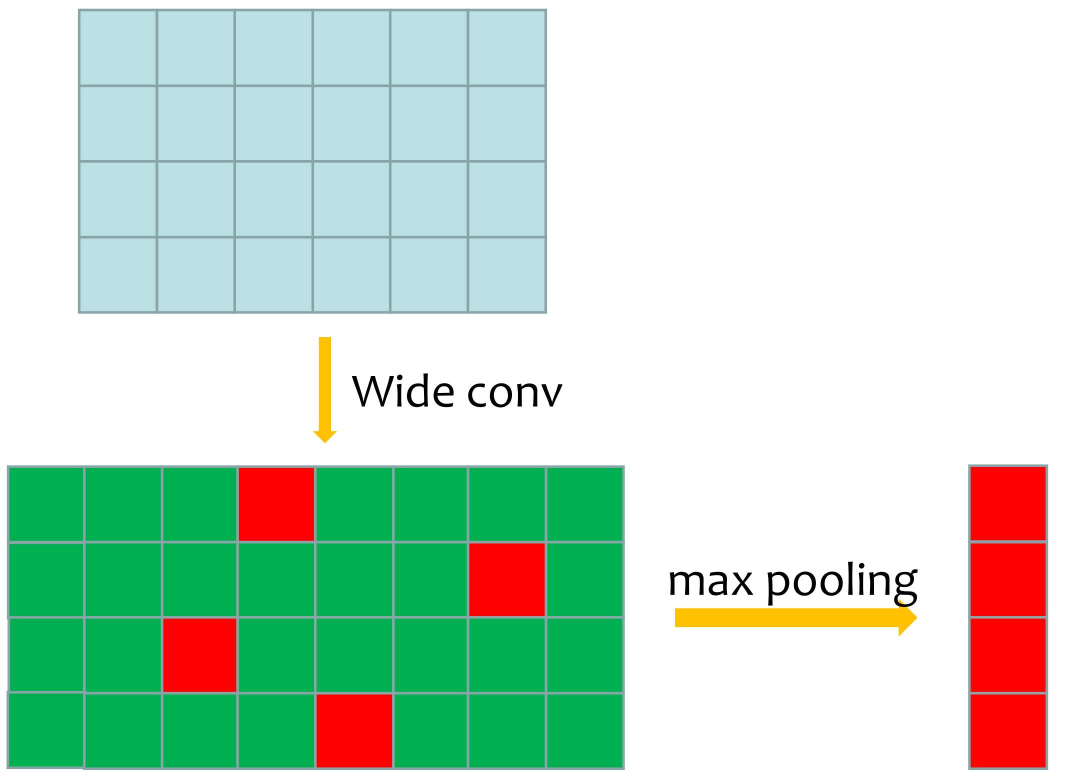
Sentence Modelling

- One dimensional convolution
  - Narrow: $m, s \Rightarrow c = s - m + 1$, where $s > m$



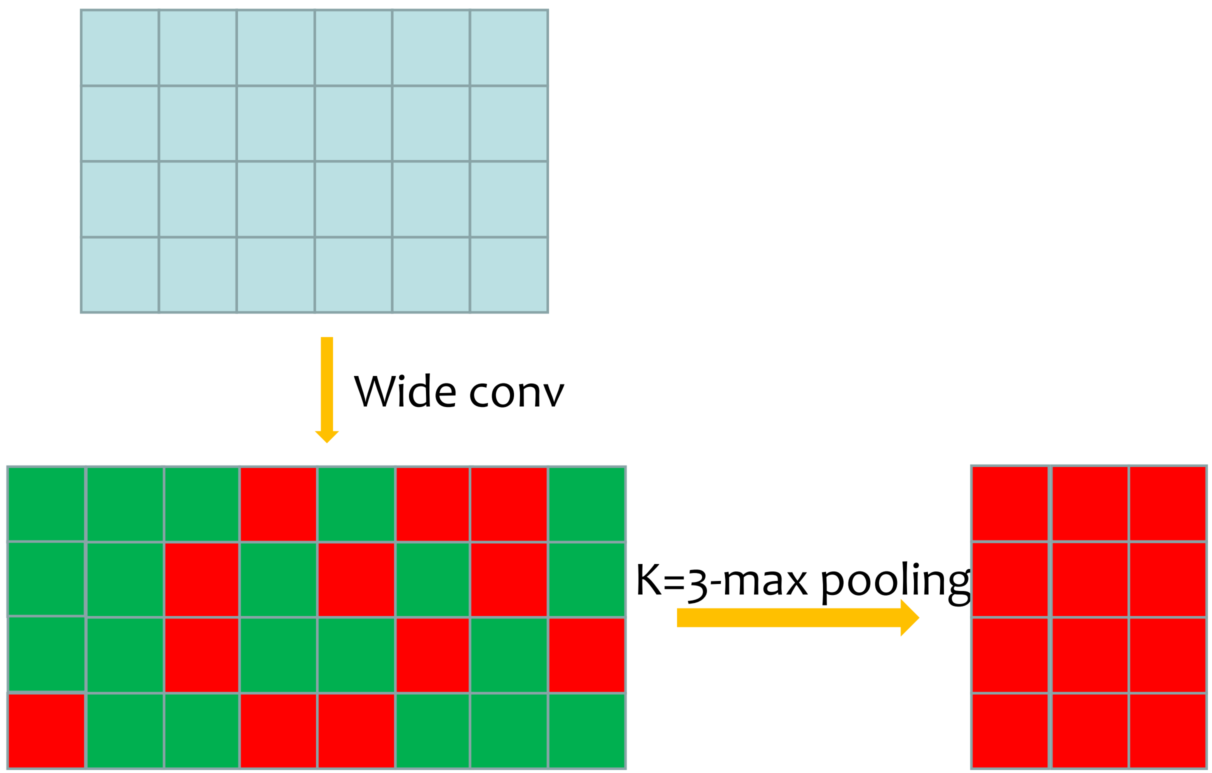
Sentence Modelling

- K-max pooling
  - Instead of pooling 1 feature along temporal dimension, we pool k most salient feature



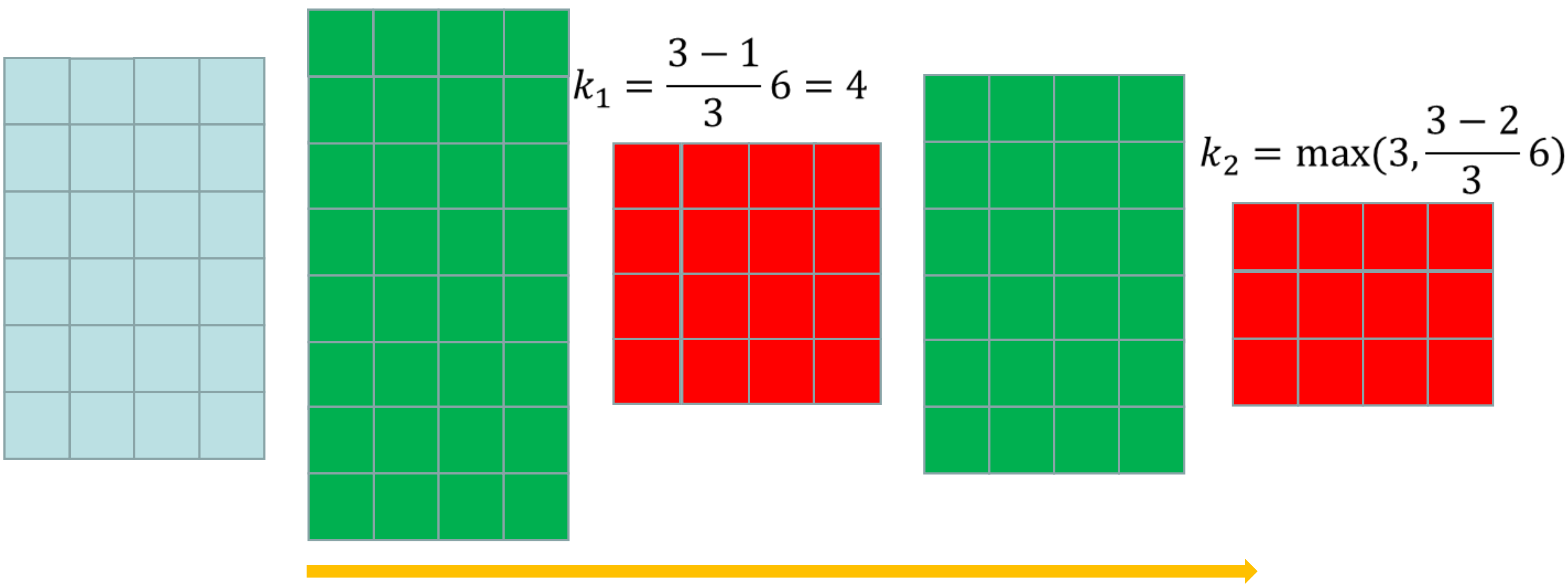
Sentence Modelling

- K-max pooling
  - Instead of pooling 1 feature along temporal dimension, we pool k most salient feature



Sentence Modelling

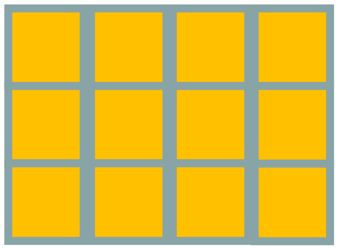
- Dynamic K-max pooling
  - $k_l = \max(k_{top}, [\frac{L-l}{L}s])$, once network's #layers determined, L and k_{top} will be determined



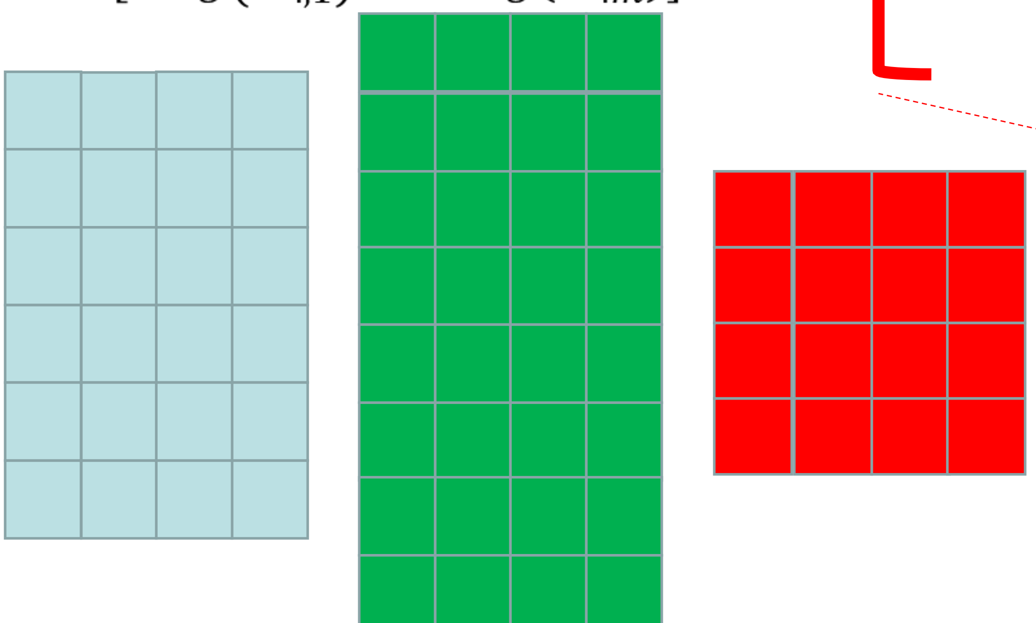
$k_{top} = 3, L = 3$

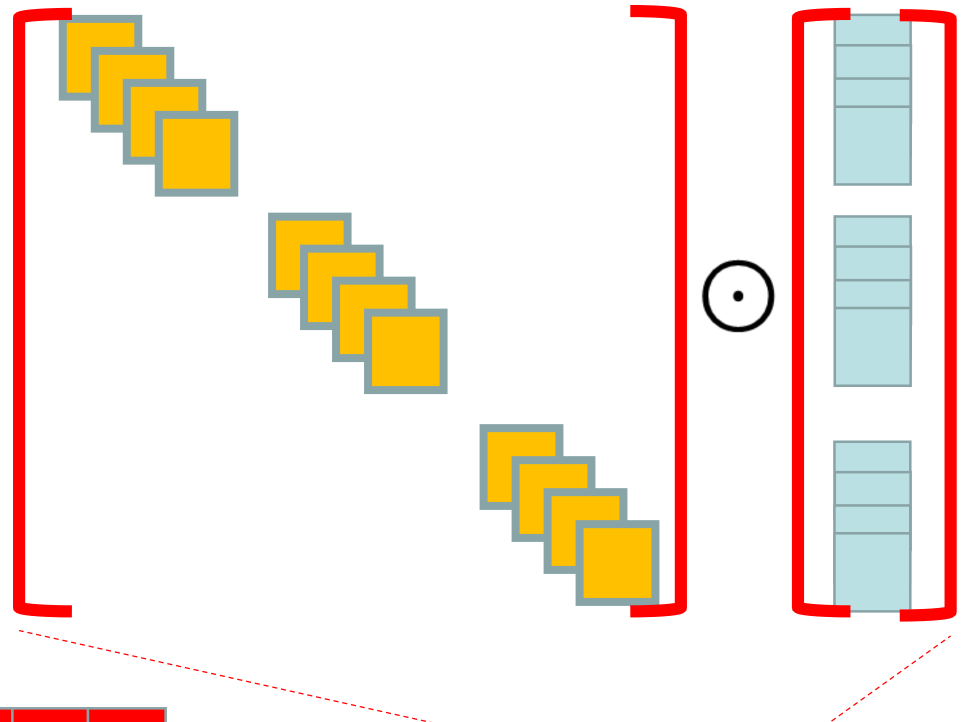
Sentence Modelling

- Nonlinear feature function



$M = [diag(\boldsymbol{m}_{:,1}), \ldots, diag(\boldsymbol{m}_{:m})]$



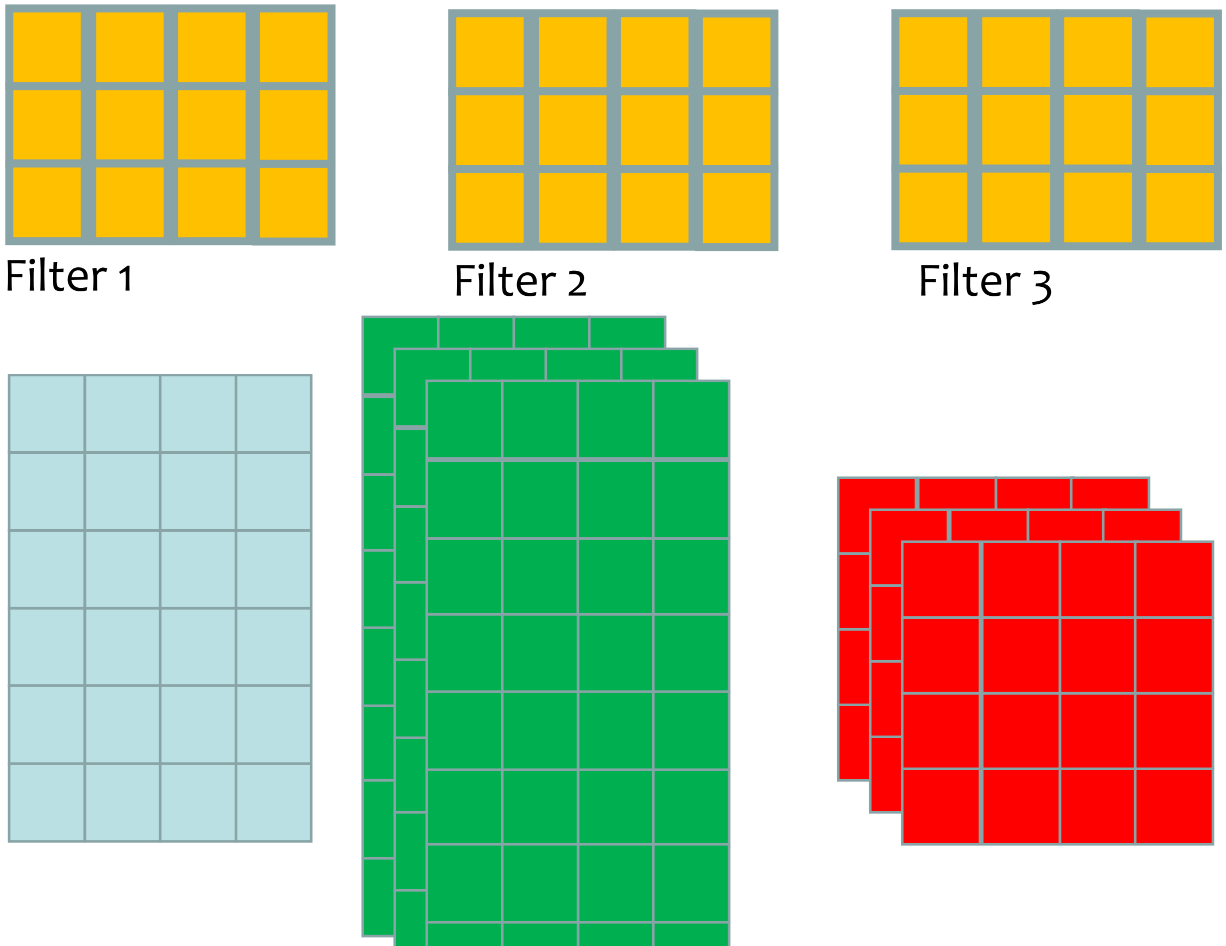


$$a = g\left(\mathbf{M}\begin{bmatrix}\mathbf{w}_j\\ \vdots \\ \mathbf{w}_{j+m-1}\end{bmatrix} + \mathbf{b}\right)$$

g is element wise

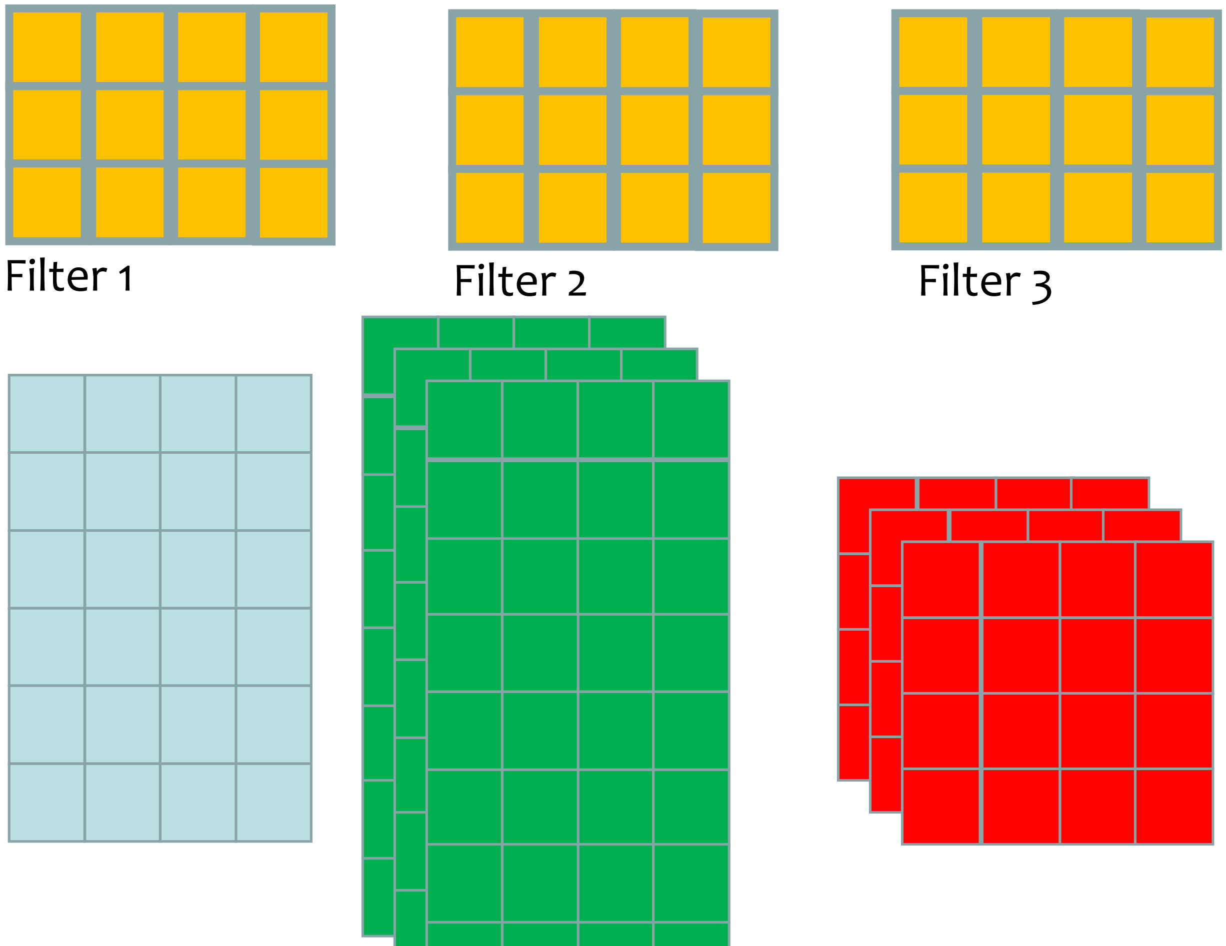
Sentence Modelling

- Multiple feature maps
  - That is we have more than one filters



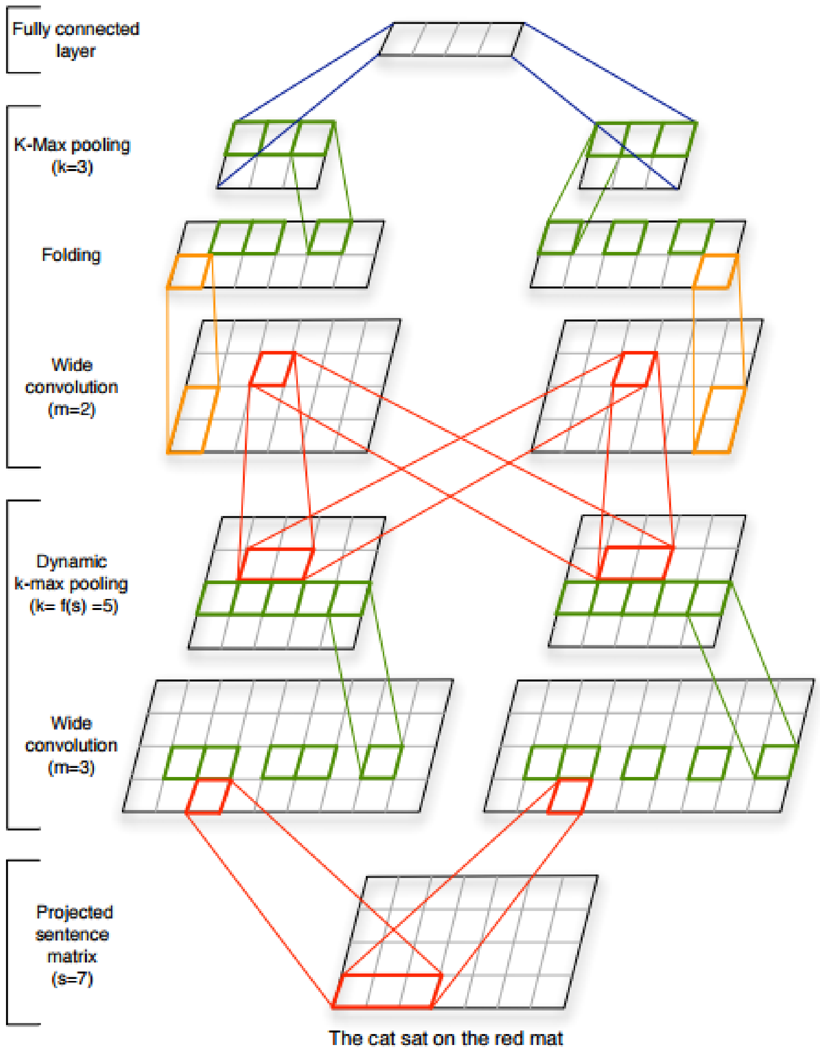
Sentence Modelling

- Dealing with variable length
  - Pooled feature map: $3(\#filter) \times 4(k) \times 4(d)$



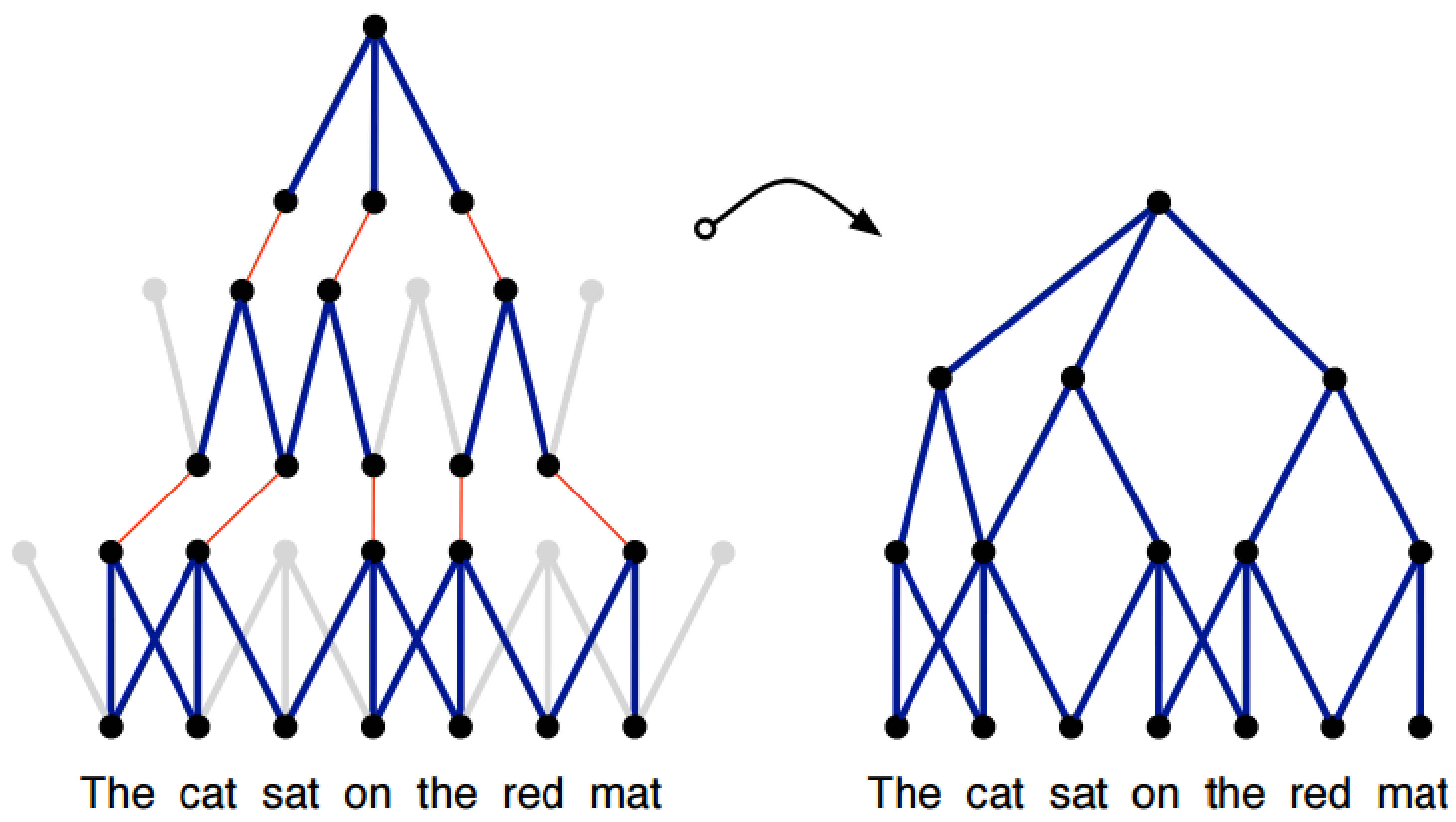
Sentence Modelling

- The whole architecture



Sentence Modelling

- Induced feature graph



Sentence Modelling II

- Convolutional neural network for paraphrase identification, NAACL 2015
- MultiGranCNN: an architecture for general matching of text chunks on multiple levels of granularity, ACL 2015
- Multi-Perspective sentence similarity modeling with convolutional neural networks, EMNLP 2015

Sentence Modelling II

- **Convolutional neural network for paraphrase identification, NAACL 2015**
  - Multi-granular interaction features

- The task: Paraphrase identification

Sentence Modelling II

Sentence Modelling II

- **MultiGranCNN: an architecture for general matching of text chunks on multiple levels of granularity, ACL 2015**
  - Multi-granular interaction features
- The task: Paraphrase identification

Sentence Modelling III

- Dependency-based convolutional neural networks for sentence embedding, ACL 2015
- Natural language inference by tree-based convolution and heuristic matching, ACL 2015
- A position encoding convolutional neural network based on dependency tree for relation classification, EMNLP 2016

Sentence Modelling III

- Dependency Sensitive Convolutional Neural Networks for Modeling Sentences and Documents, NAACL 2016
- The forest convolutional network: compositional distributional semantics with a neural chart and without binarization, EMNLP 2015

Sentence Pair Modelling

- **Dual linguistic spans modelling**
- It concerns multi-granular relationships between two linguistic or even multimodal information carrier spans
  - **Phrases in two sentence**
  - **Two sentences or super sentences**
    - Premise, conclusion; two adjacent text span with discourse relation
    - Reading material with question, answer candidates
  - **Cross-lingual**
    - Machine translation pairs
  - **Cross-modal**
    - Image(video), caption pairs
    - Image(video) with question, answer candidates

(Multi-)Semantic Units Modeling

- Event detection and domain adaptation with convolutional neural networks
- Event extraction via dynamic multi-pooling convolutional neural networks
- Modeling skip-gram for event detection with convolutional neural networks, EMNLP 2016

(Multi-)Semantic Units Modeling

- Speculation and negation scope detection via convolutional neural networks, EMNLP 2016
- Intra-sentential subject zero anaphora resolution using multi-column convolutional neural network, EMNLP 2016

(Multi-)Semantic Units Modeling

- Question Answering over Freebase with multi-column convolutional neural networks, ACL 2015
- Capturing semantic similarity for entity linking with convolutional neural networks, NAACL 2016

Document & Text Modelling

- Effective use of word order for text categorization with convolutional neural network, NAACL 2015
- Non-linear text regression with a deep convolutional neural network, ACL 2015

Ranking

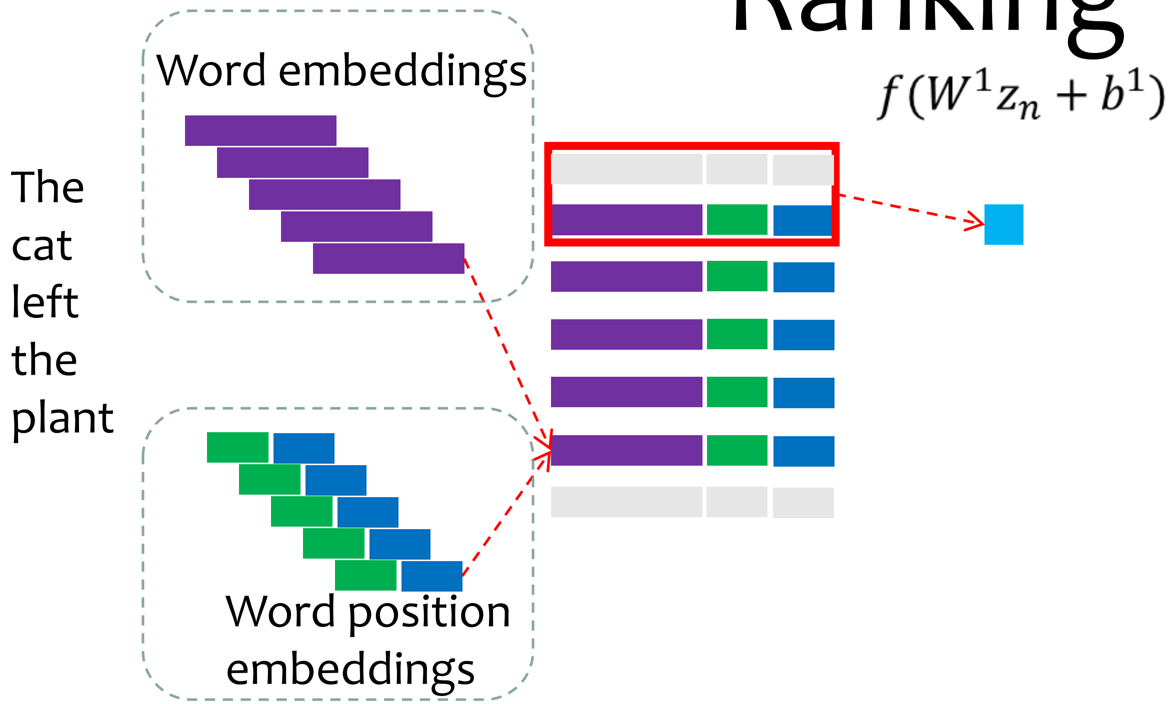
- A re-ranking model for dependency parser with recursive convolutional neural network, ACL 2015
- Classifying Relation by Ranking with Convolutional Neural Networks, ACL 2015

Ranking

- **Classifying Relation by Ranking with Convolutional Neural Networks, ACL 2016**
  - Embed every symbolic item!

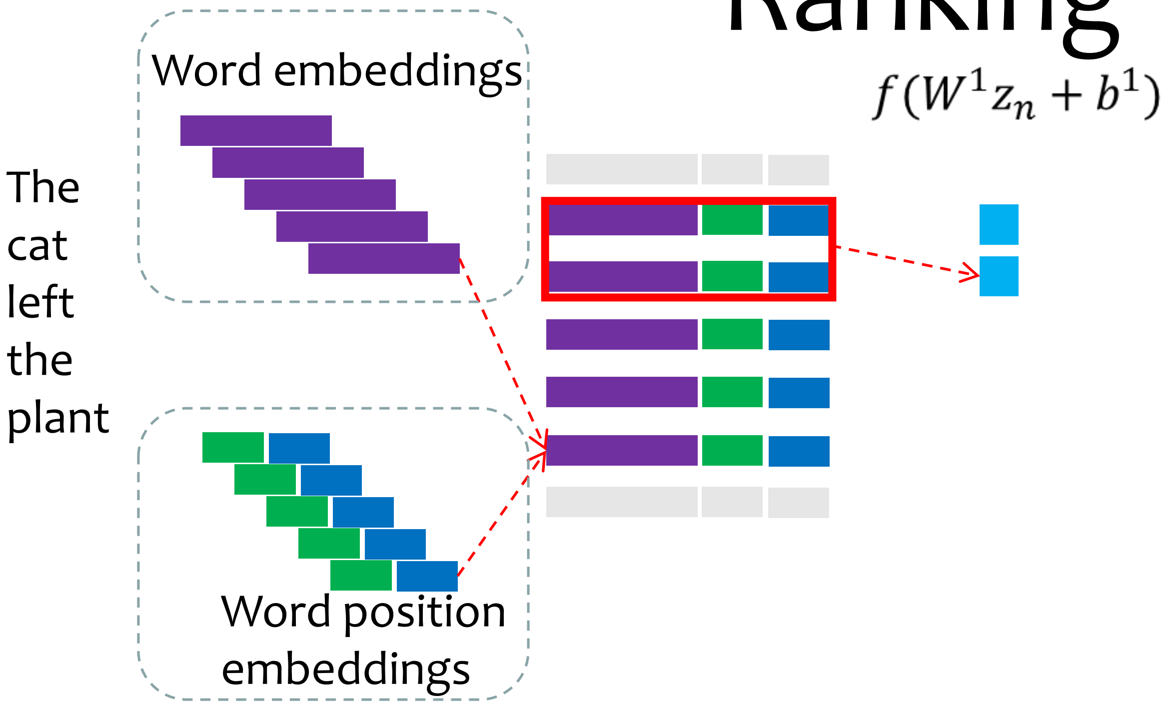
- What is relation classification?
  - SemEval-2010 Task 8, 10717 annotate, 9 relations
  - Supervision signal
    - *The [car] left the [plant]. r=Content-container*
  - Prediction
    - *The [introduction]$_{e1}$ in the [book]$_{e2}$ is a summary of what is in the text. ⇒ r? e1, e2*

Ranking



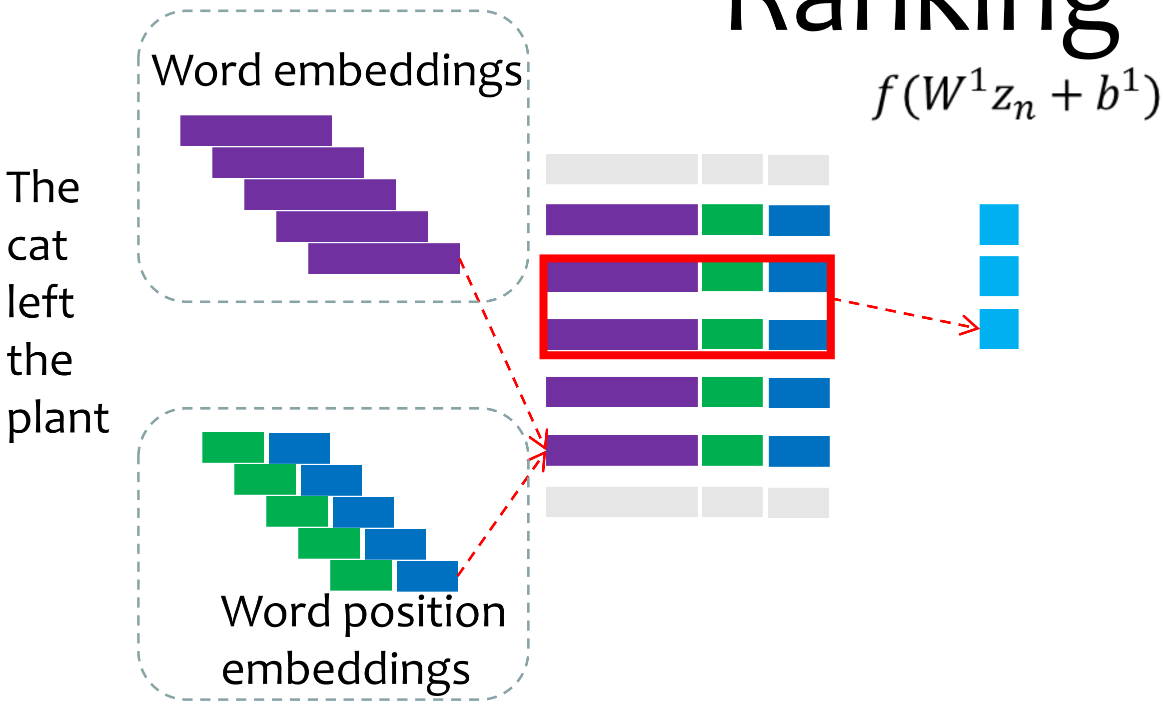
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



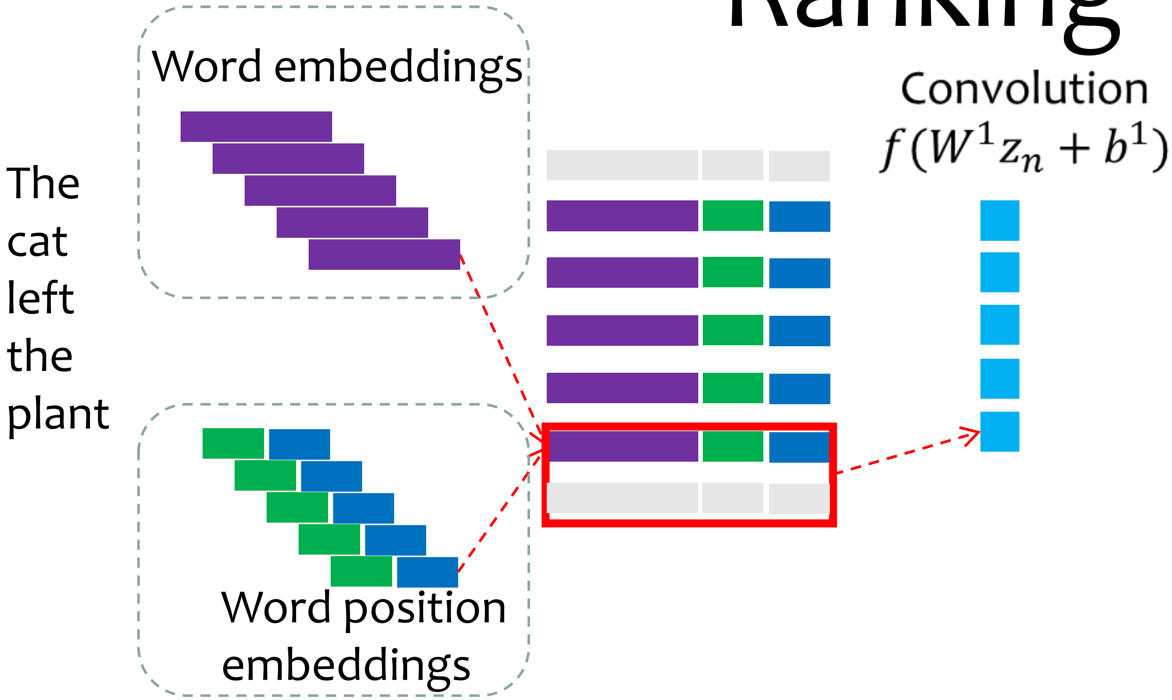
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



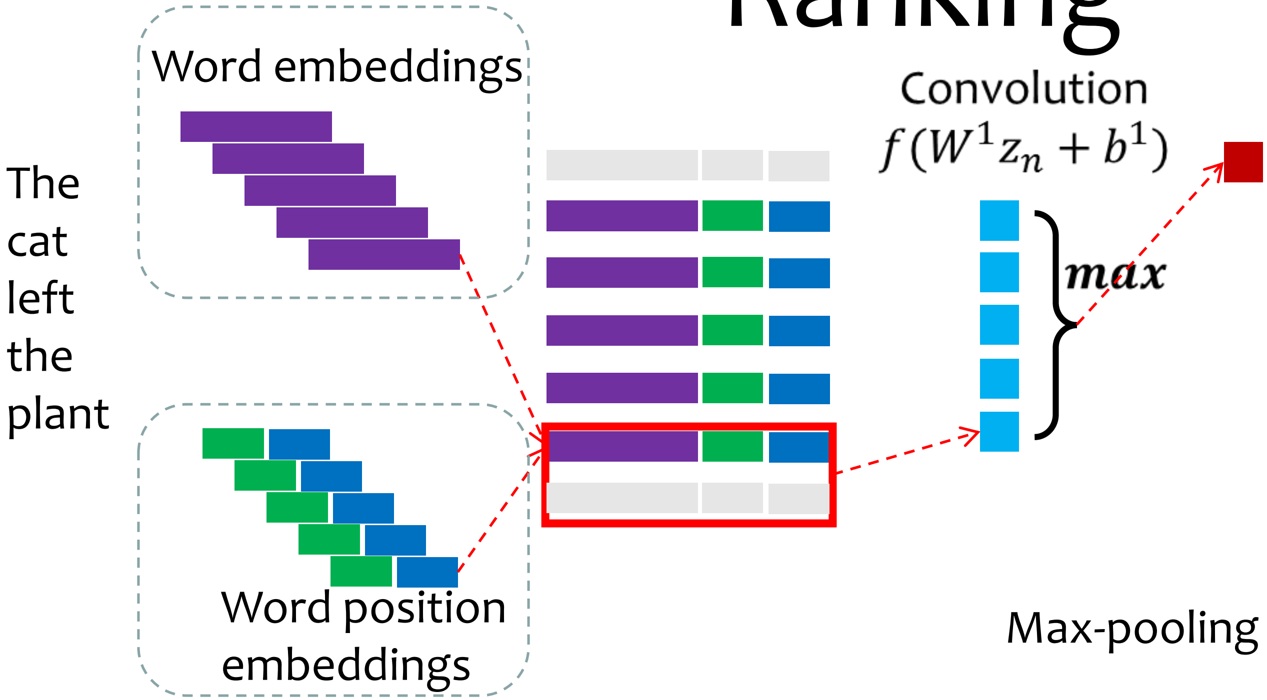
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



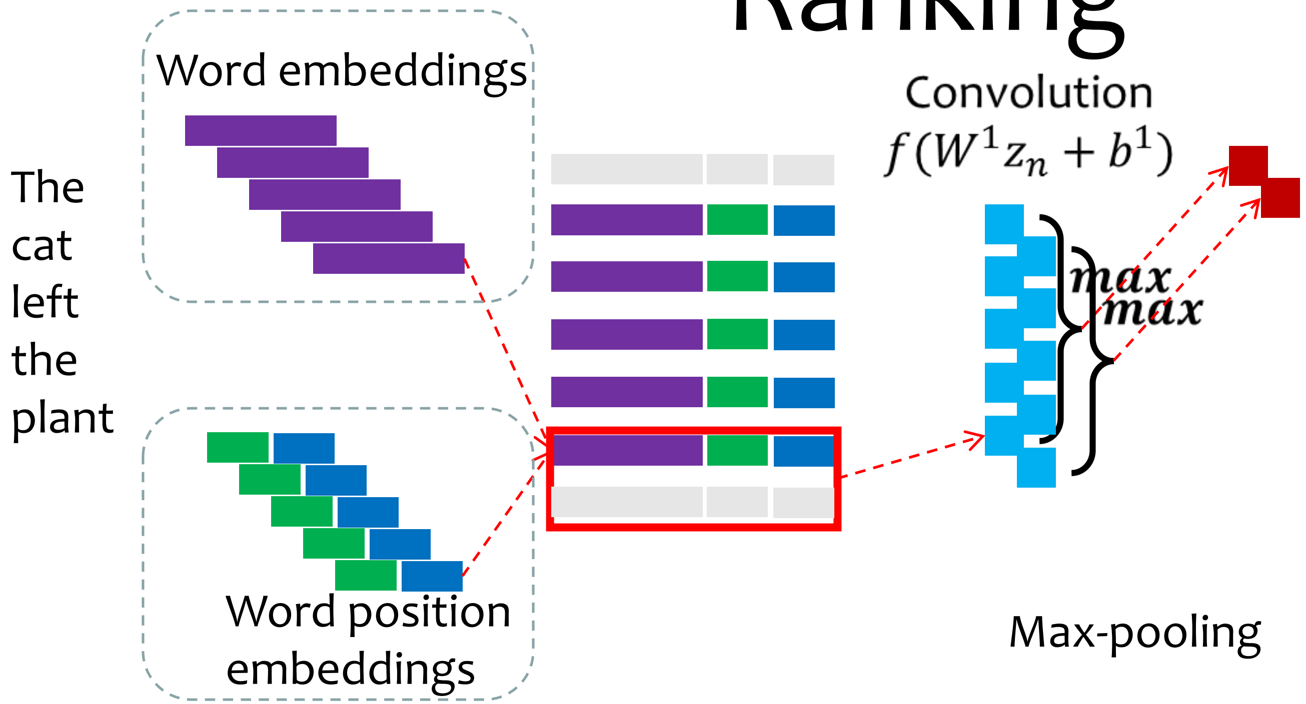
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



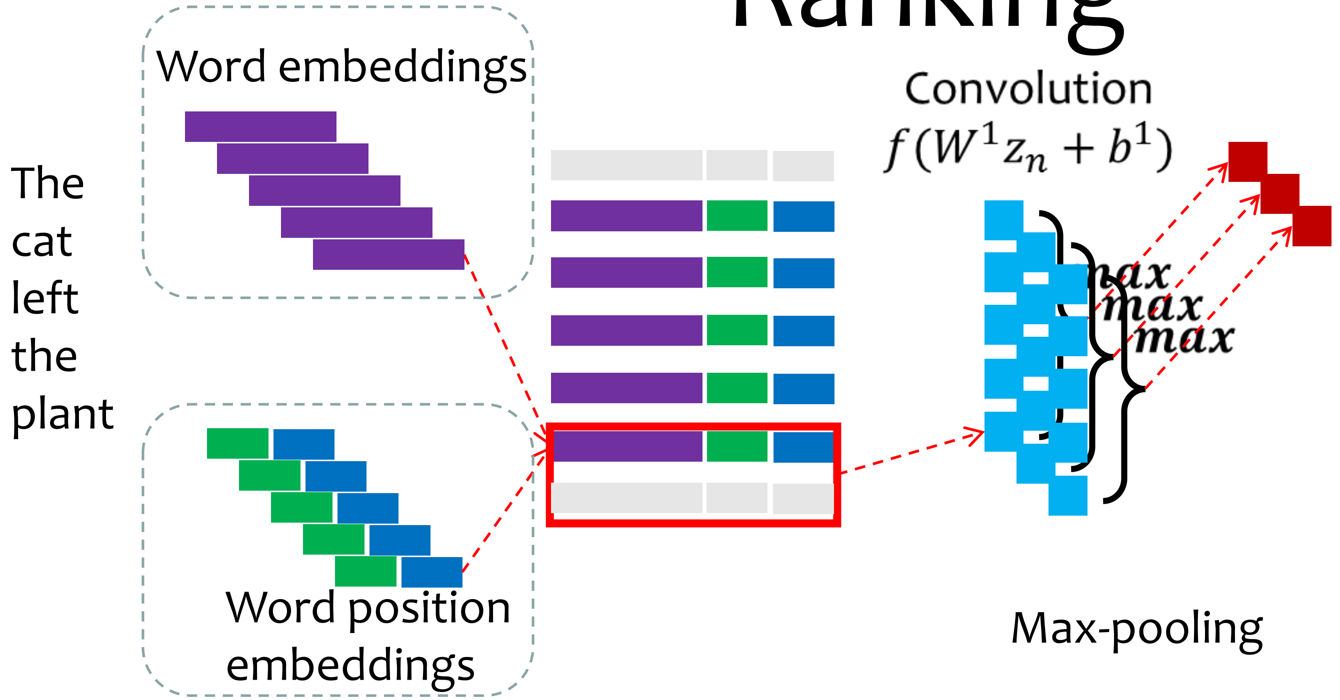
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



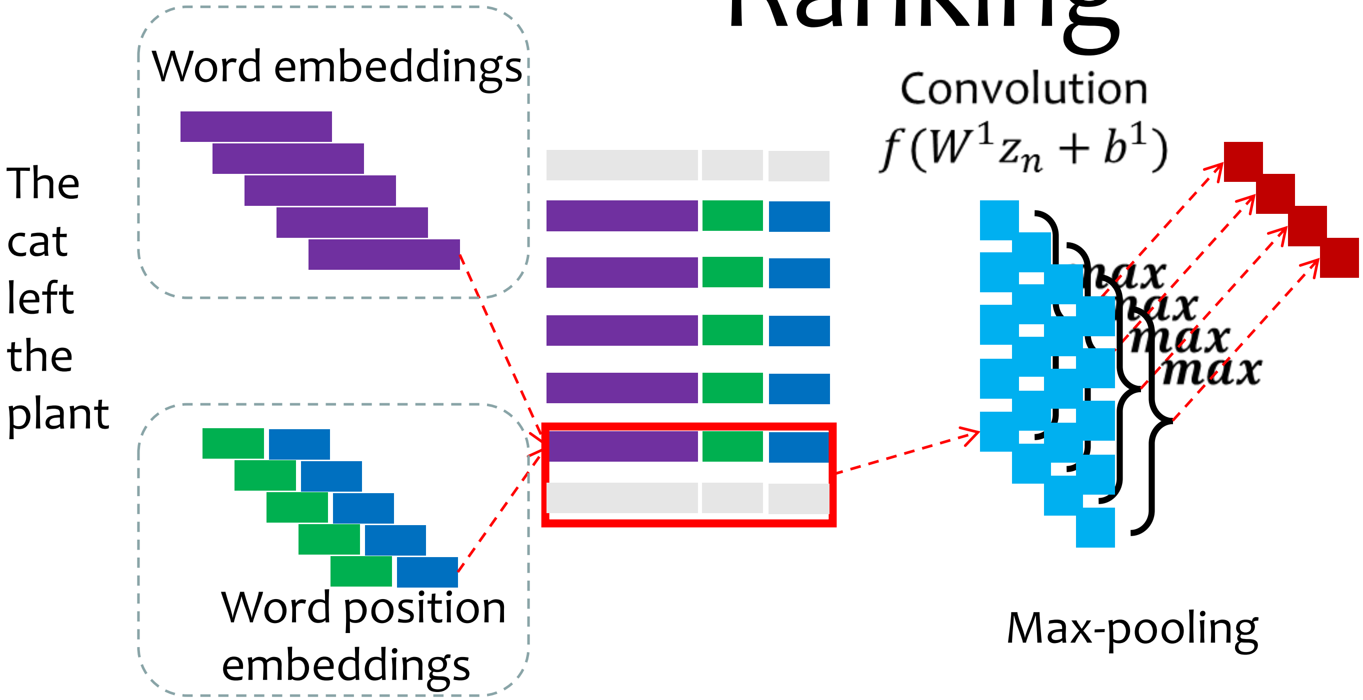
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



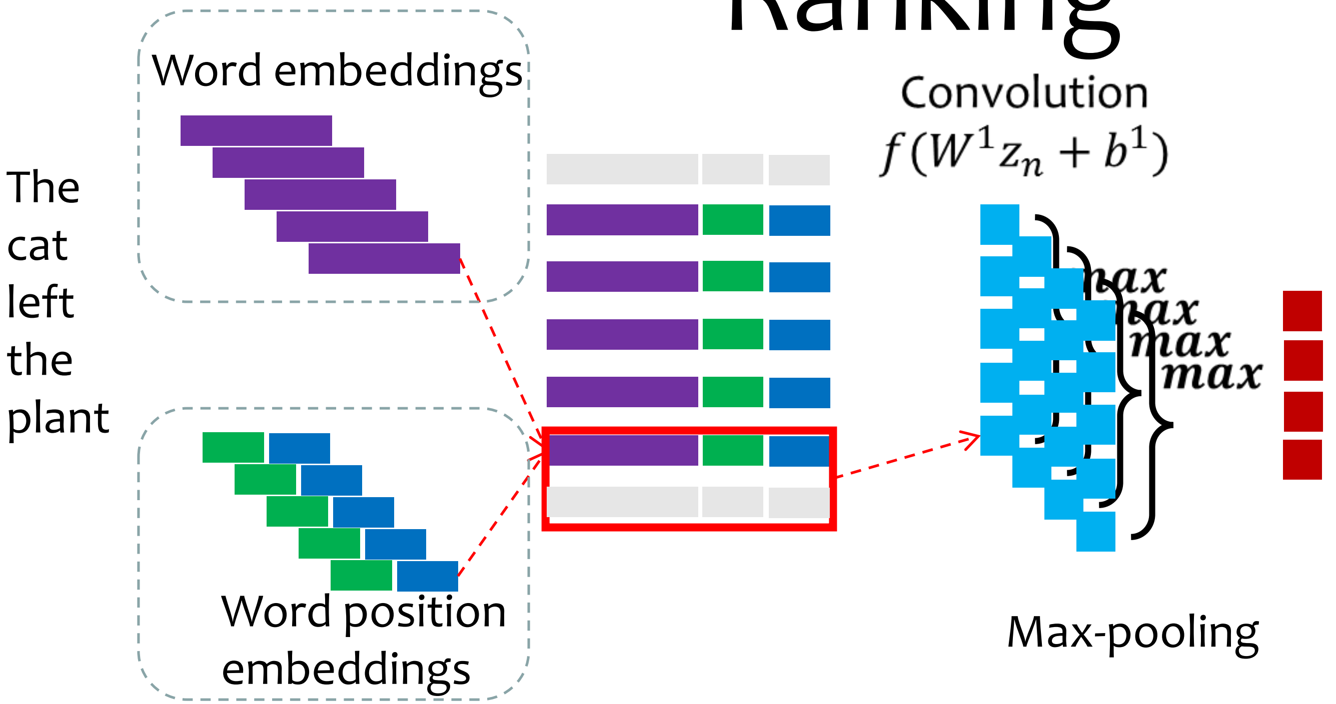
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



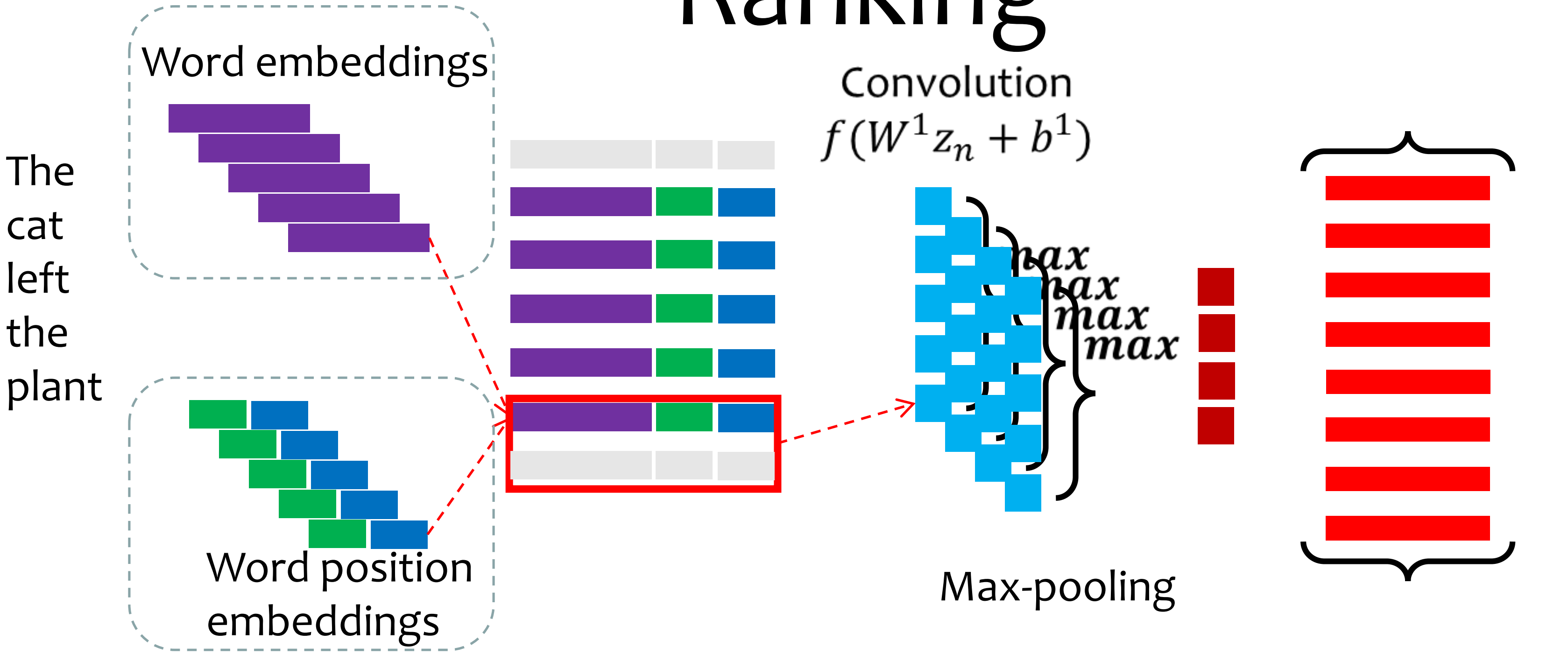
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



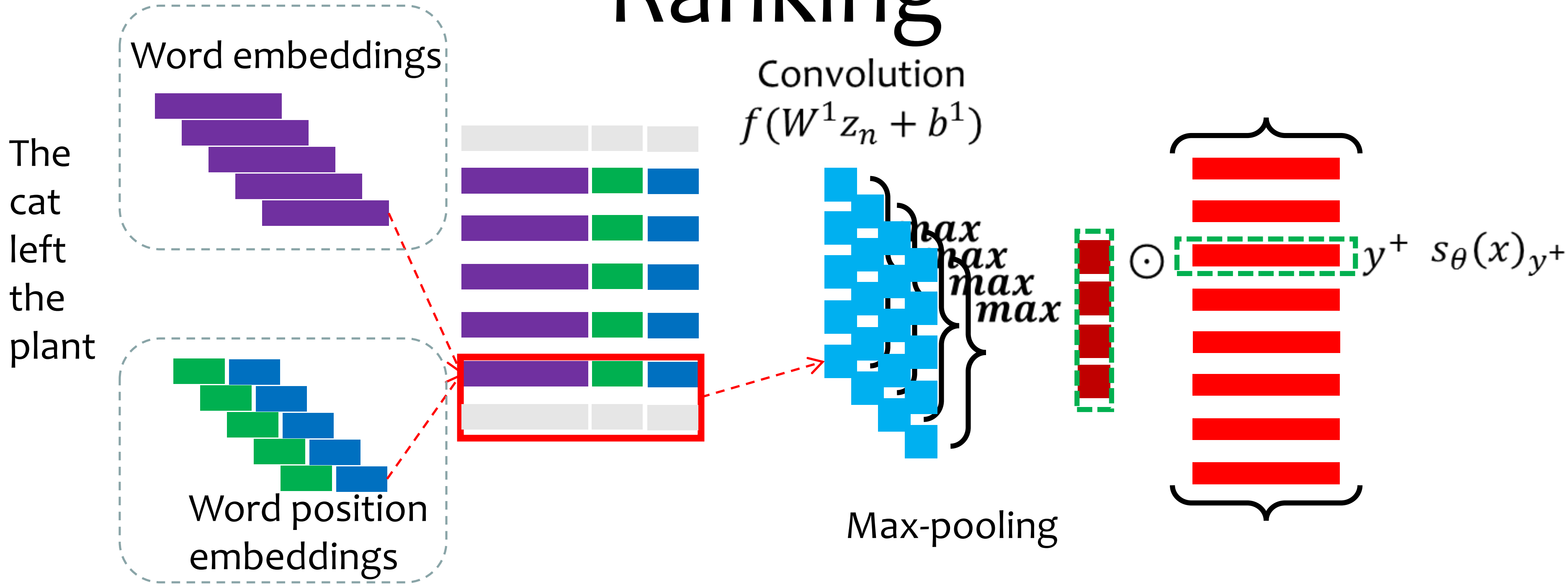
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



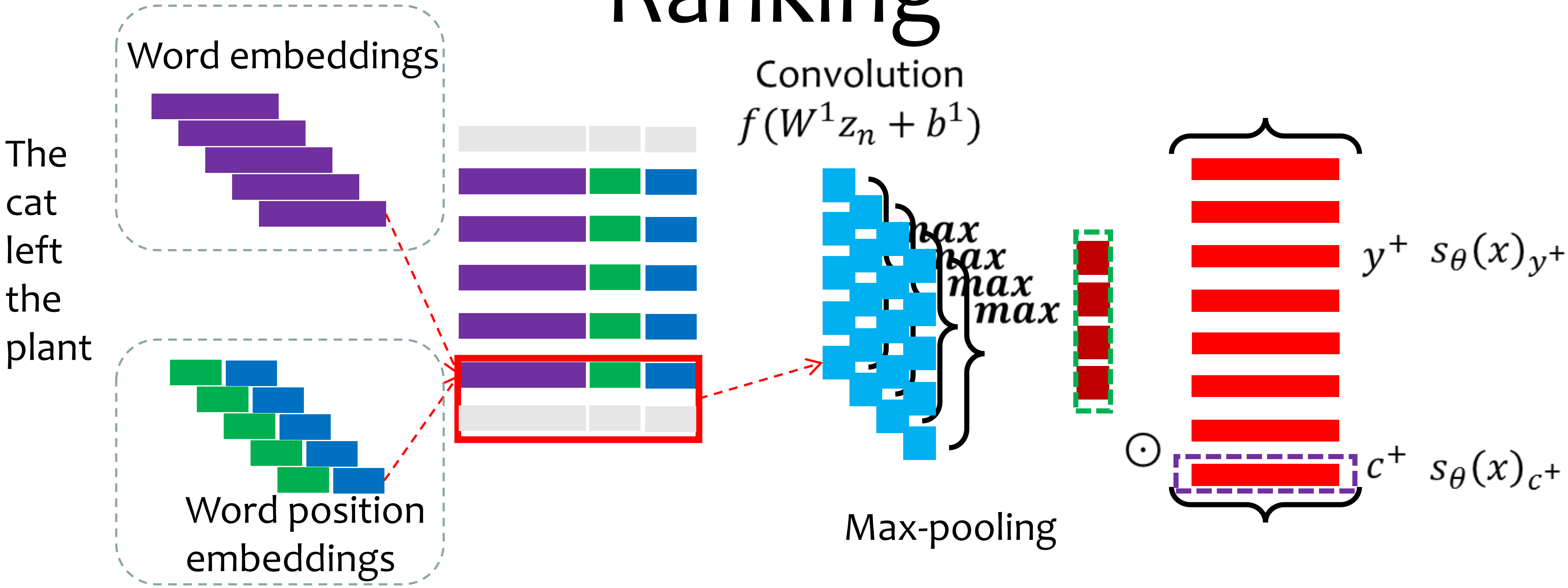
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



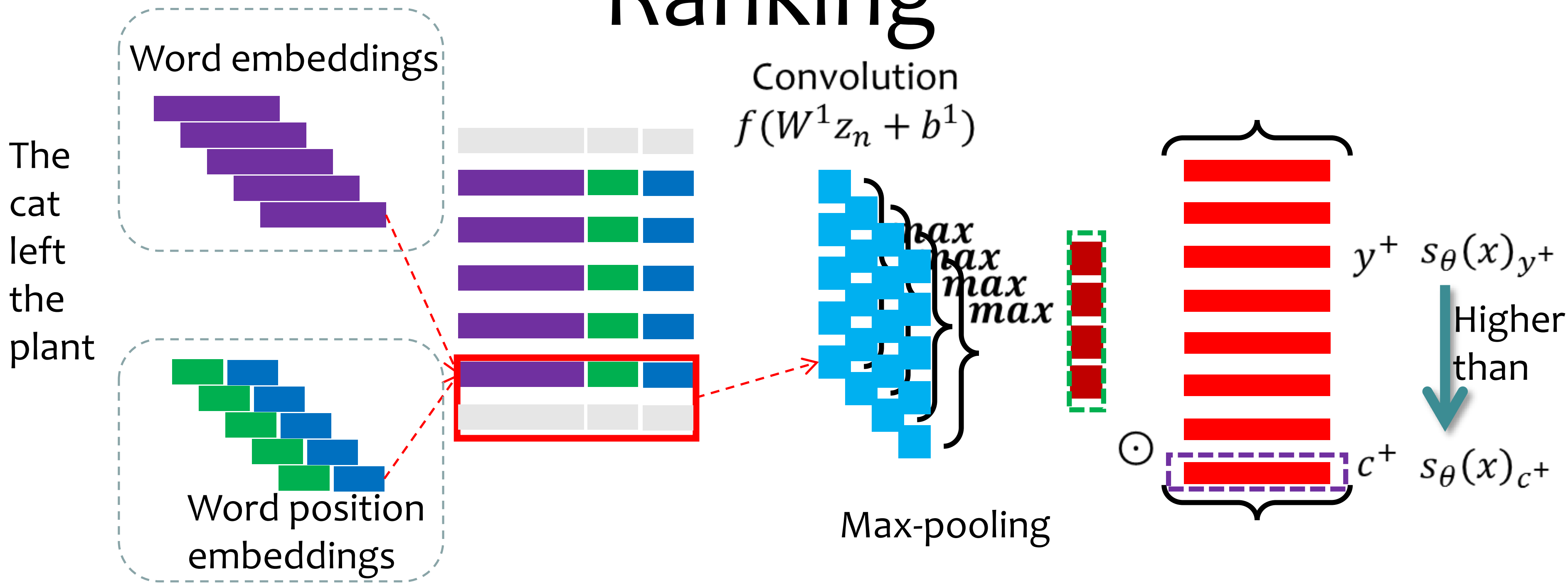
- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking



- Embed each word w_i in x, x is the sentence
- Embed each word's position w.r.t. target nouns
- Embed each semantic relation with $W_c^{classes}$

Ranking

- Ranking loss during training
  - We get our sentence representation r_x
  - We can compute score of each relation
  - We know the ground truth y^+, so does its score $s_\theta(x)_{y^+}$
  - We find the a class $c^- \neq y^+$ with highest score
  - Our loss is
  - $L = \log\left(1 + \exp\left(\gamma(m^+ - \boldsymbol{s_\theta(x)_{y^+}})\right)\right) + \log(1 + \exp(\gamma(m^- + \boldsymbol{s_\theta(x)_{c^-}})))$

Structure Prediction I

- Probabilistic graph-based dependency parsing with convolutional neural network, ACL 2016
- Exploring convolutional and recurrent neural networks in sequential labelling for dialogue topic tracking, ACL 2016

Structure Prediction II

- Probabilistic graph-based dependency parsing with convolutional neural network, ACL 2016
- Exploring convolutional and recurrent neural networks in sequential labelling for dialogue topic tracking, ACL 2016

Tricks and Philosophy

- Gating mechanism
- Attention mechanism
- Composition on structure

Gating Mechanism

- Semi-supervised question retrieval with gated convolutions, NAACL 2016
- Training very deep networks, NIPS 2015 Poster Spotlight Session

Attention Mechanism

- Relation classification via multi-level attention CNNs, ACL 2016
- ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs, TACL 2016