Social Media & Text Analysis

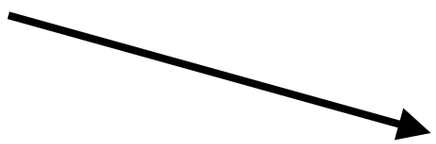
lecture 11 - Sentiment Analysis: Convolutional Neural Networks and Attention

CSE 5539-0010 Ohio State University
Instructor: Wei Xu
Website: socialmedia-class.org

slides are from Yang Yi, Graham Neubig, Richard Socher

An Example Prediction Problem: Sentence Classification

I hate this movie

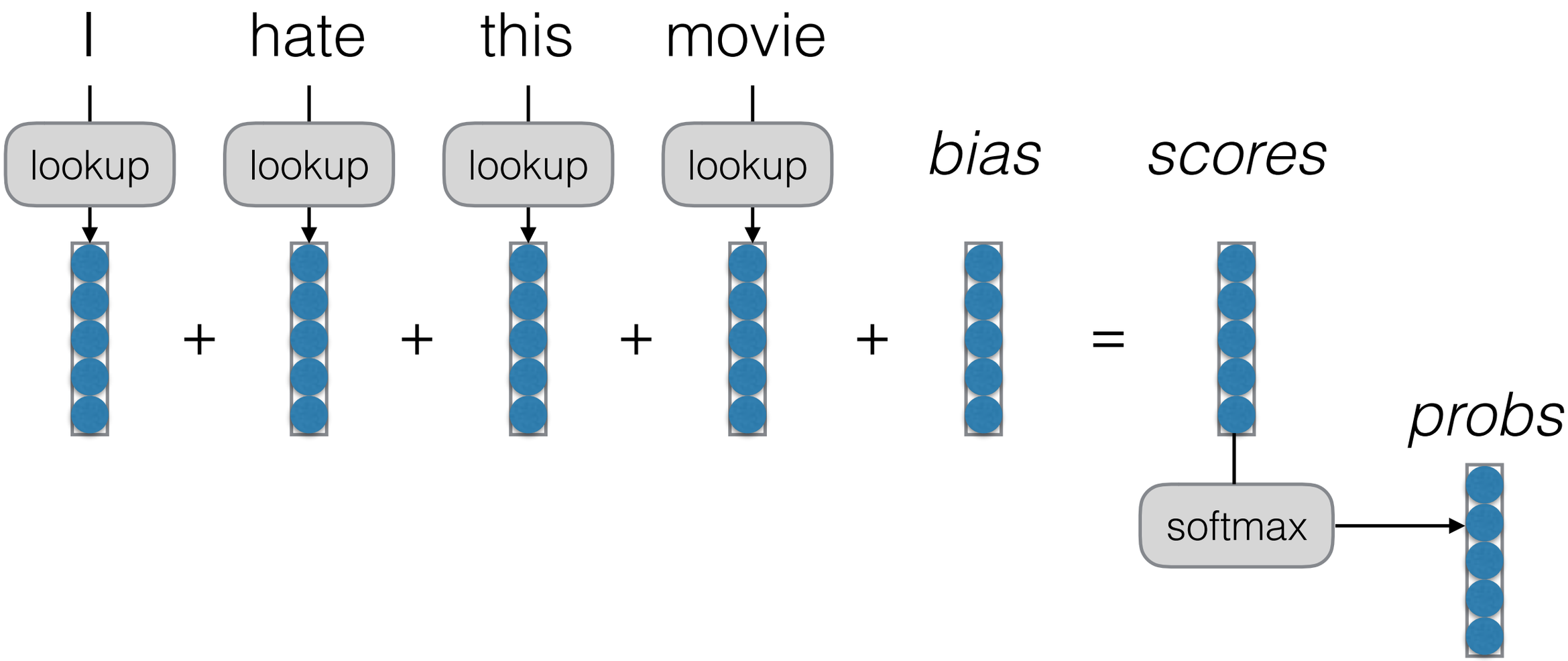


very good
good
neutral
bad
very bad

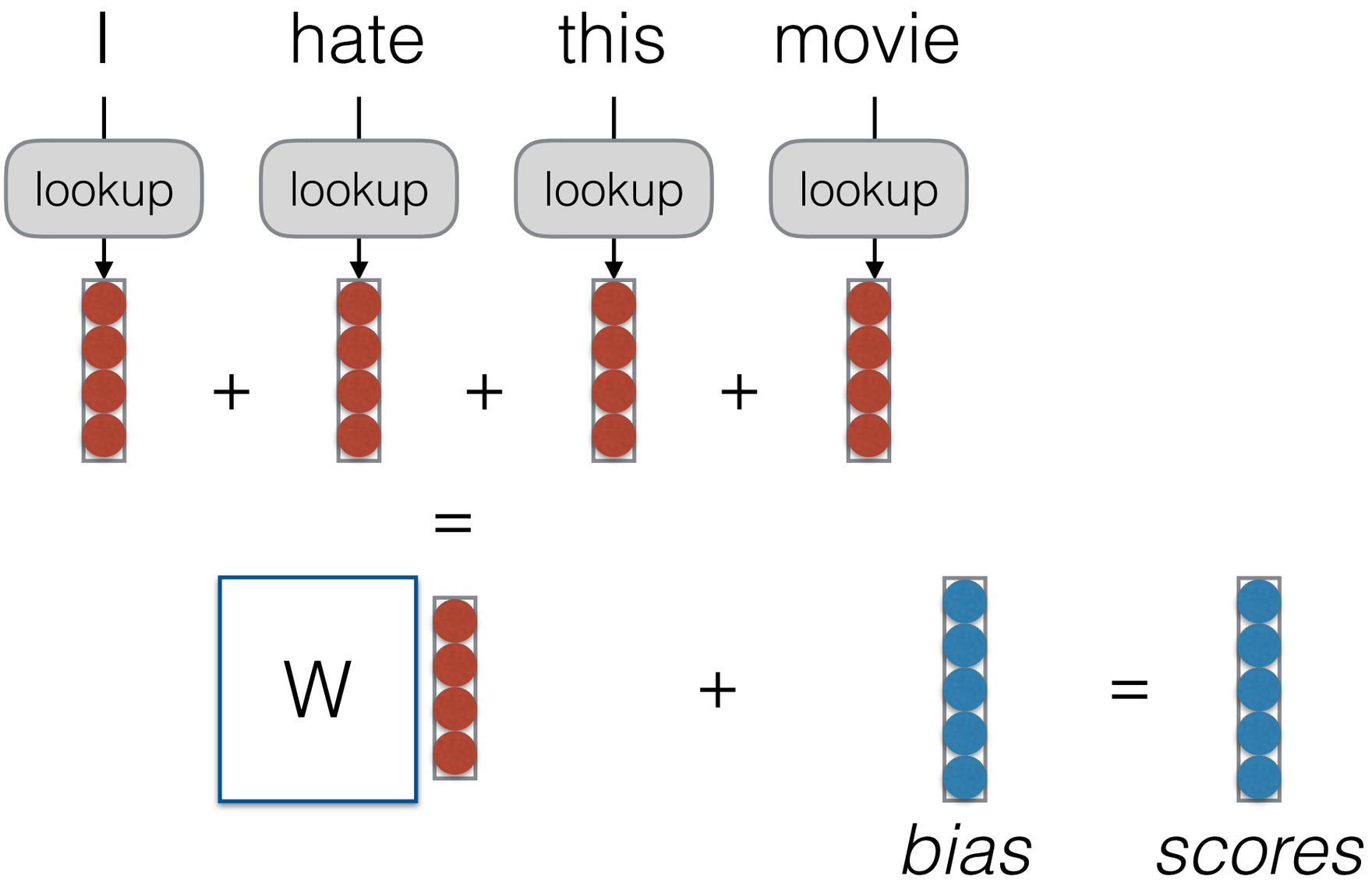
I love this movie

very good
good
neutral
bad
very bad

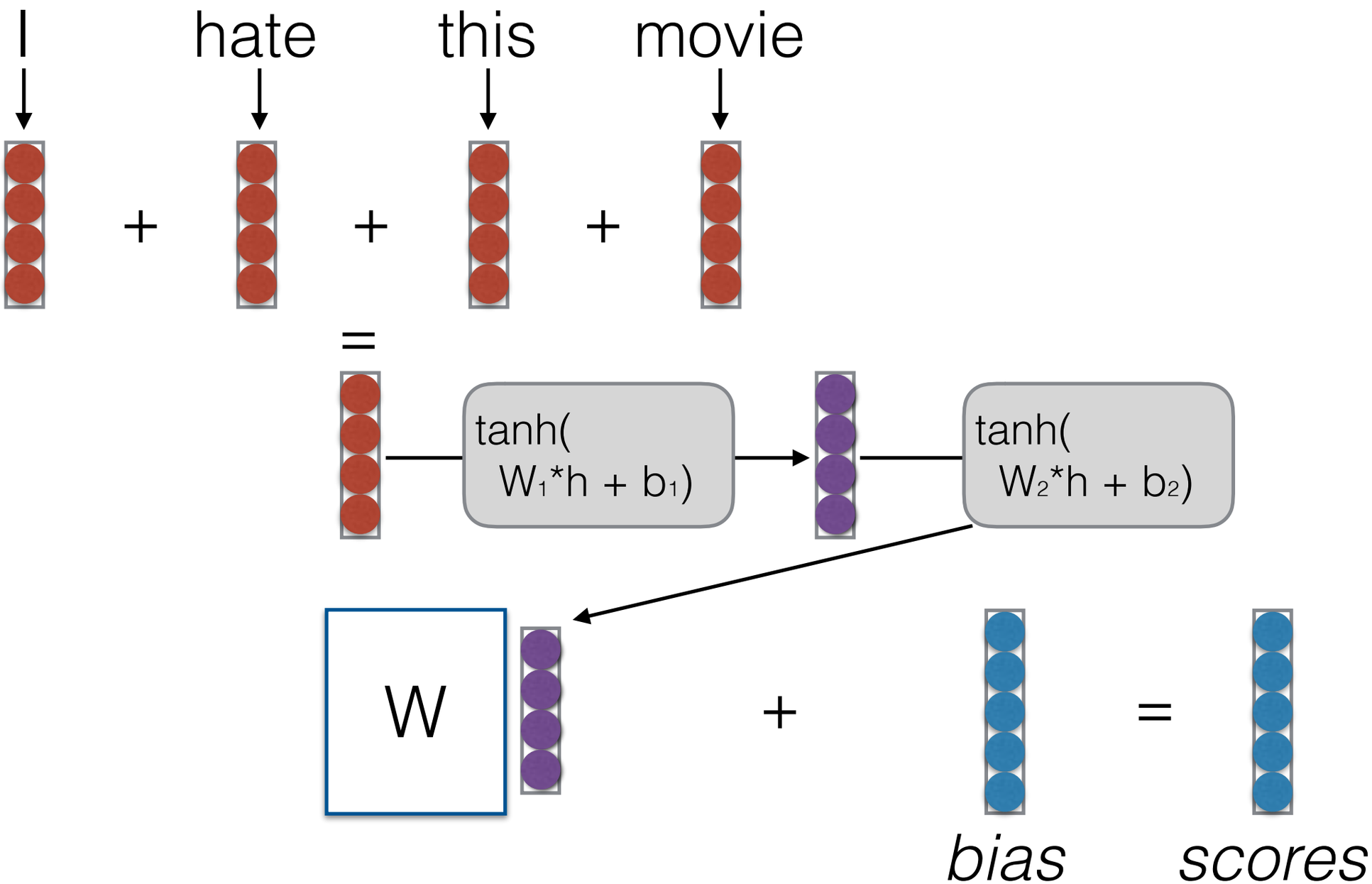
A First Try: Bag of Words (BOW)



Continuous Bag of Words (CBOW)



Deep CBOW

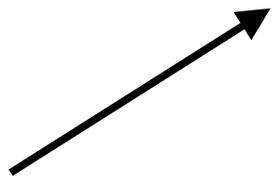


Build It, Break It

I don't love this movie

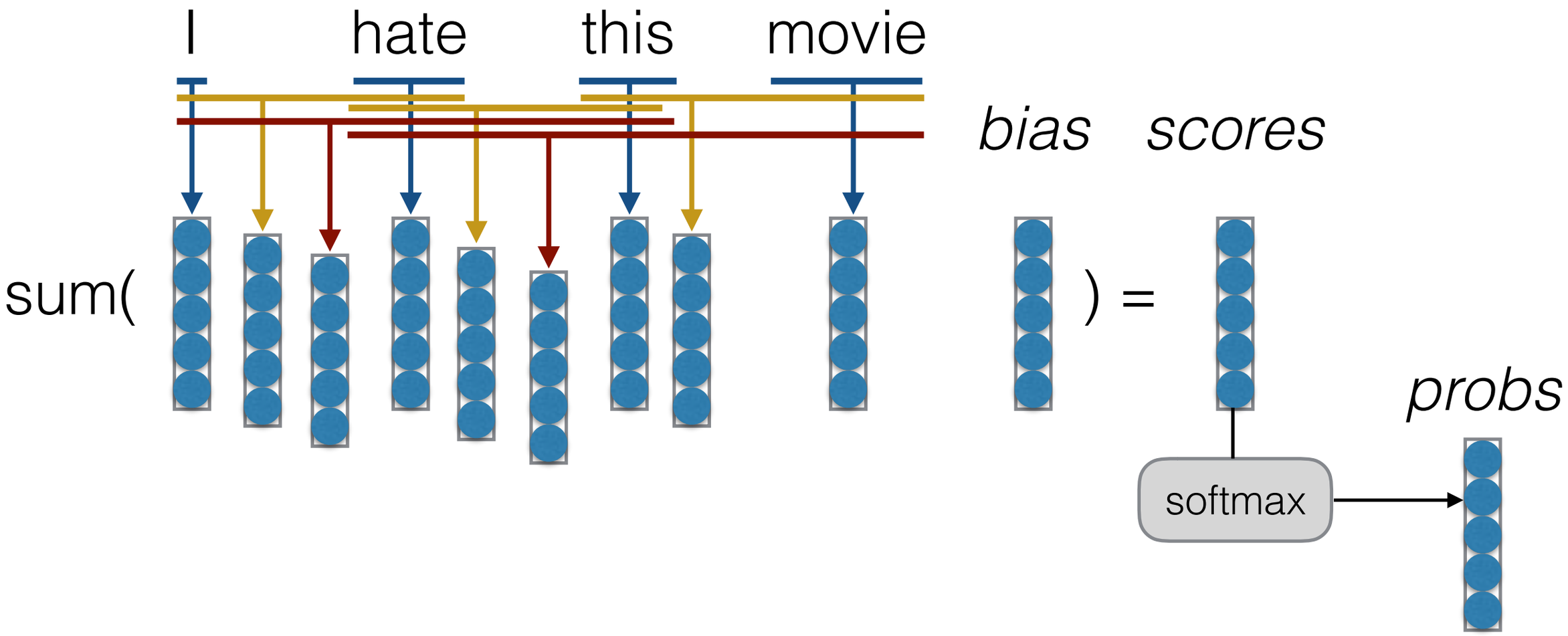
very good
good
neutral
bad
very bad

There's nothing I don't love about this movie



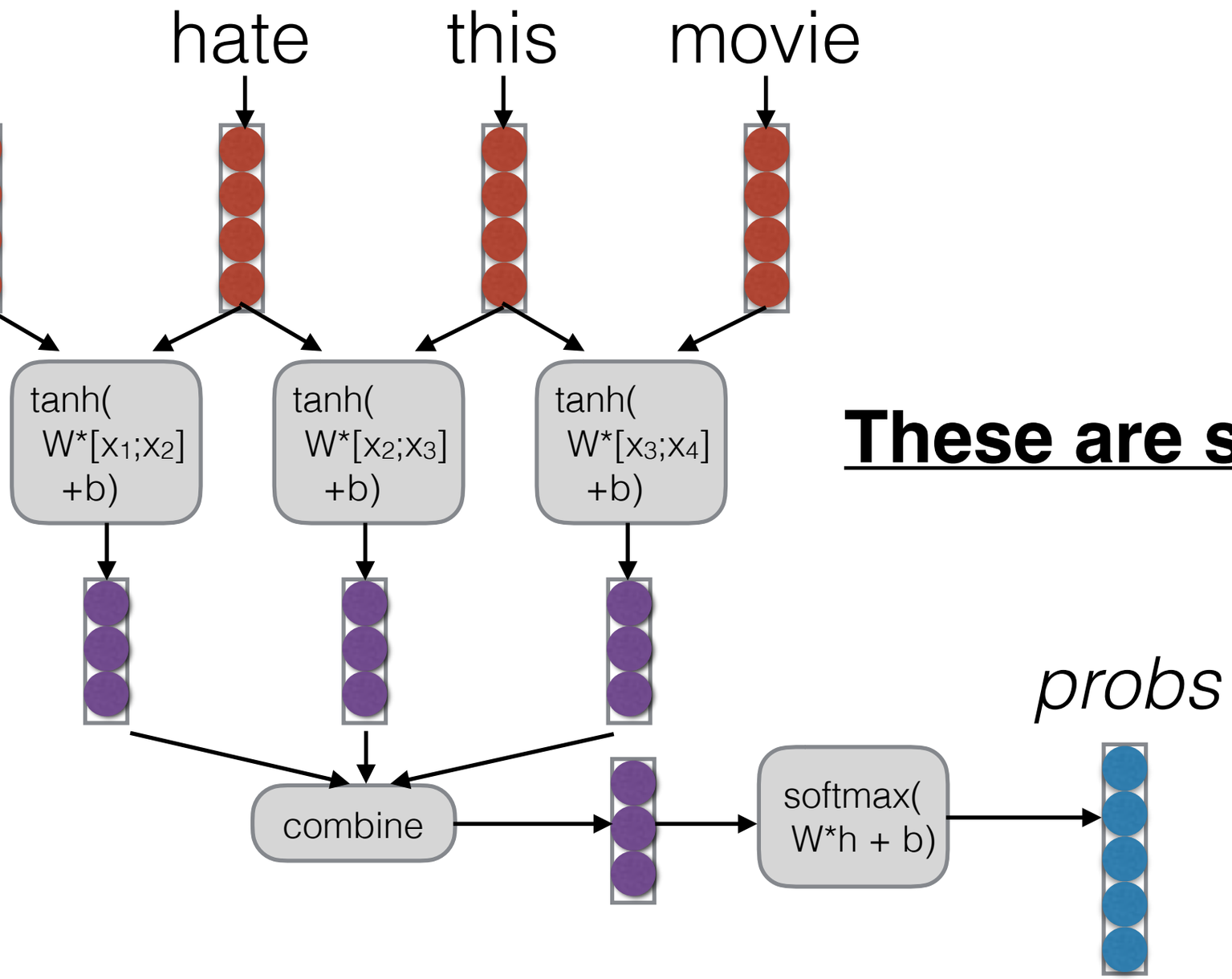
very good
good
neutral
bad
very bad

Bag of n-grams



Time Delay Neural Networks

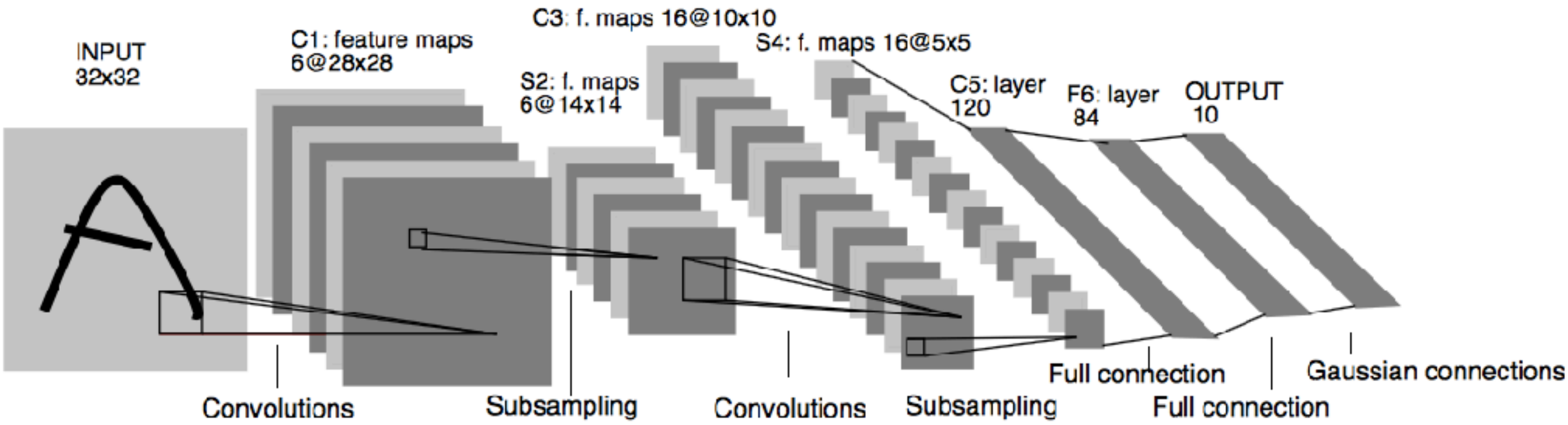
(Waibel et al. 1989)



These are soft 2-grams!

Convolutional Networks

(LeCun et al. 1997)



Parameter extraction performs a 2D sweep, not 1D

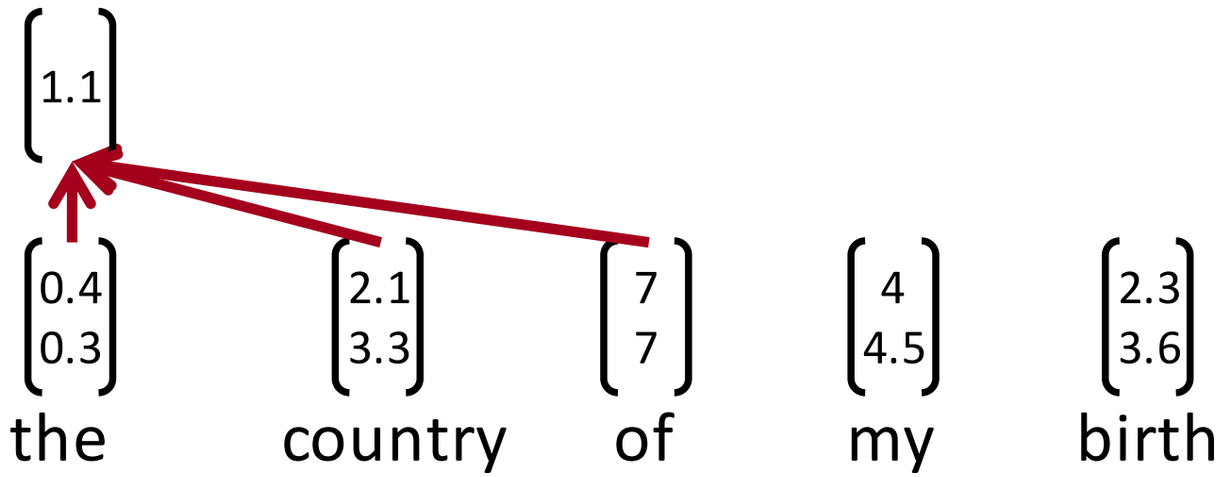
CNNs for Text
(Collobert and Weston 2011)

- 1D convolution ≈ Time Delay Neural Network
    - But often uses terminology/functions borrowed from image processing
- Two main paradigms:
    - **Context window modeling:** For tagging, etc. get the surrounding context before tagging
    - **Sentence modeling:** Do convolution to extract n-grams, pooling to combine over whole sentence

Single layer CNN

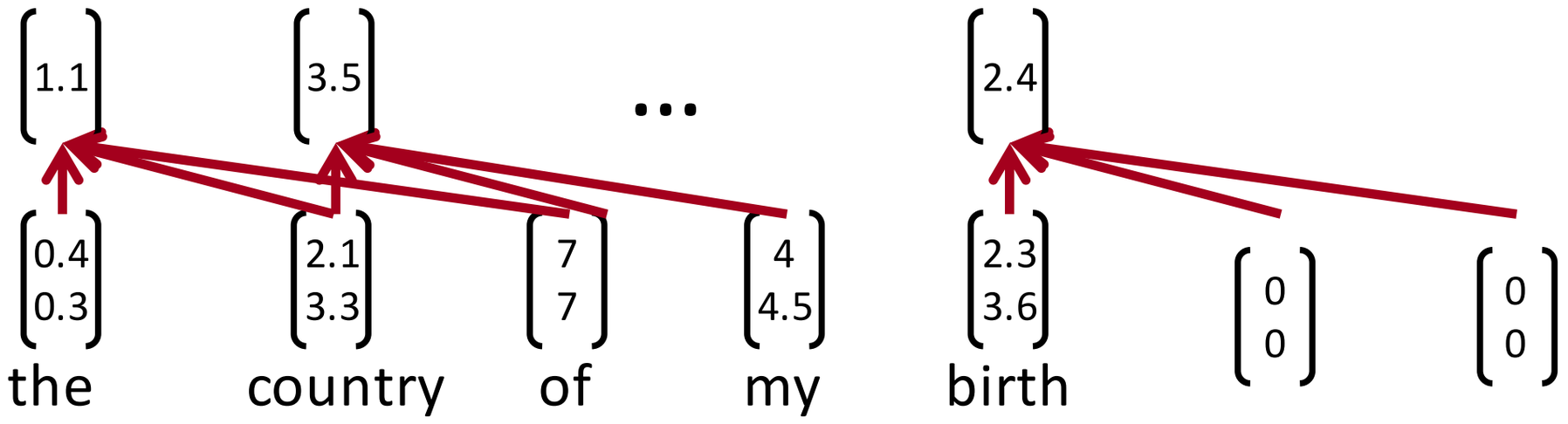
- Convolutional filter: $\mathbf{w} \in \mathbb{R}^{hk}$ (goes over window of h words)
- Note, filter is vector!
- Window size h could be 2 (as before) or higher, e.g. 3:
- To compute feature for CNN layer:

$$c_i = f(\mathbf{w}^T \mathbf{x}_{i:i+h-1} + b)$$



Single layer CNN

- Filter w is applied to all possible windows (concatenated vectors)
- Sentence: $\mathbf{x}_{1:n} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \ldots \oplus \mathbf{x}_n$
- All possible windows of length h: $\{\mathbf{x}_{1:h}, \mathbf{x}_{2:h+1}, \ldots, \mathbf{x}_{n-h+1:n}\}$
- Result is a feature map: $\mathbf{c} = [c_1, c_2, \ldots, c_{n-h+1}] \in \mathbb{R}^{n-h+1}$



Single layer CNN: Pooling layer

- New building block: Pooling
- In particular: max-over-time pooling layer
- Idea: capture most important activation (maximum over time)

- From feature map $\mathbf{c} = [c_1, c_2, \ldots, c_{n-h+1}] \in \mathbb{R}^{n-h+1}$

- Pooled single number: $\hat{c} = \max\{\mathbf{c}\}$

- But we want more features!

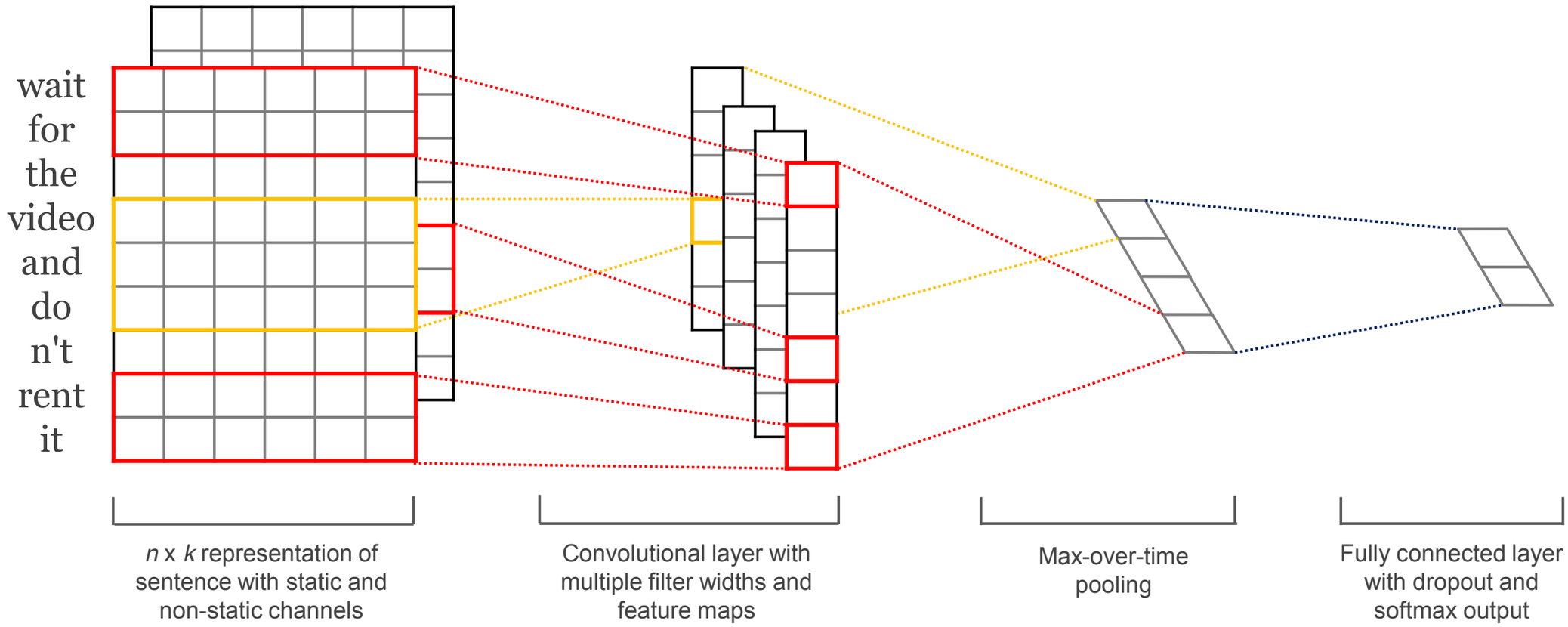
Solution: Multiple filters

- Use multiple filter weights w
- Useful to have different window sizes h
- Because of max pooling $\hat{c} = \max\{\mathbf{c}\}$, length of **c** irrelevant

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

- So we can have some filters that look at unigrams, bigrams, tri-grams, 4-grams, etc.

Figure from Kim (2014)



Tricks to make it work better: Dropout

- Idea: randomly mask/dropout/set to 0 some of the feature weights z
- Create masking vector r of Bernoulli random variables with probability p (a hyperparameter) of being 1

- Delete features during training:

$$y = softmax\left(W^{(S)}(r \circ z) + b\right)$$

- Reasoning: Prevents co-adaptation (overfitting to seeing specific feature constellations)

Tricks to make it work better: Dropout

$$y = softmax\left(W^{(S)}(r \circ z) + b\right)$$

- At training time, gradients are backpropagated only through those elements of z vector for which $r_i = 1$

- At test time, there is no dropout, so feature vectors z are larger.
- Hence, we scale final vector by Bernoulli probability p

$$\hat{W}^{(S)} = pW^{(S)}$$

- Kim (2014) reports **2 – 4% improved accuracy** and ability to use very large networks without overfitting

All hyperparameters in Kim (2014)

- Find hyperparameters based on dev set
- Nonlinearity: reLu
- Window filter sizes h = 3,4,5
- Each filter size has 100 feature maps
- Dropout p = 0.5
- L2 constraint s for rows of softmax s = 3
- Mini batch size for SGD training: 50
- Word vectors: pre-trained with word2vec, k = 300

- During training, keep checking performance on dev set and pick highest accuracy weights for final evaluation

A Case Study

Overcoming Language Variation in Sentiment Analysis with Social Attention

Yi Yang
Bloomberg LP

Work performed at Georgia Tech with Jacob Eisenstein.

Language variation in sentiment analysis

"I would like to believe he's **sick** rather than just mean and evil."

"You could've been getting down to this **sick** beat."

Language variation in sentiment analysis

I am sick and weak

THIS IS SO SICK THANK U

Language variation in sentiment analysis

I am sick and weak

THIS IS SO SICK THANK U

Language variation in sentiment analysis

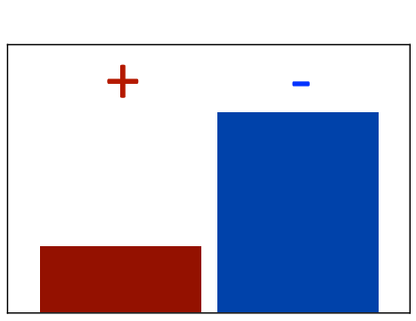
I am sick and weak

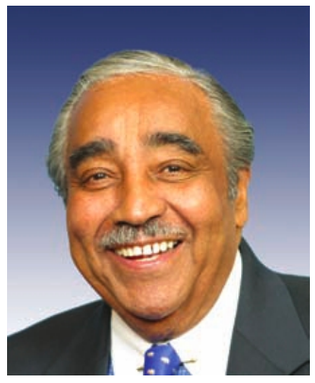
THIS IS SO SICK THANK U

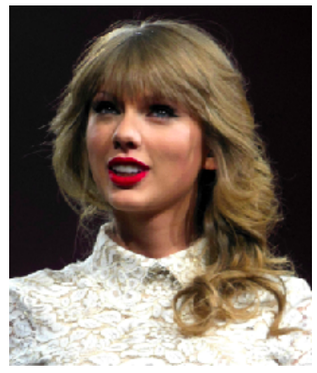
ALEX THIS IS SO SICK

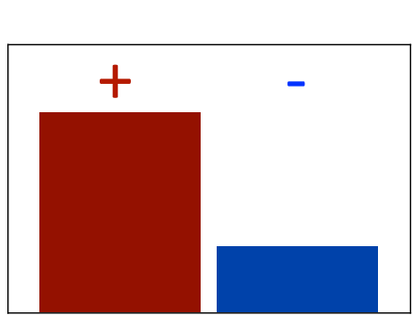
Personalized sentiment analysis

- **Goal**: personalized conditional likelihood, $p(y|\mathbf{x}, a)$.
  - $\mathbf{x}$ is the text, and a is the author.



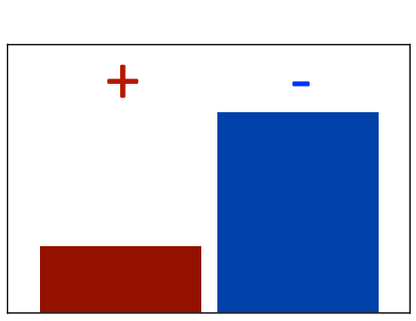


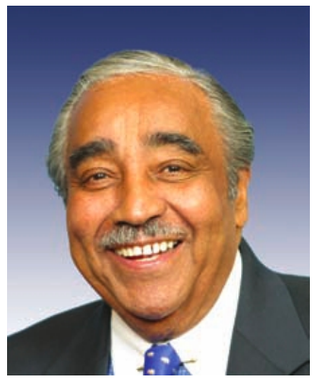


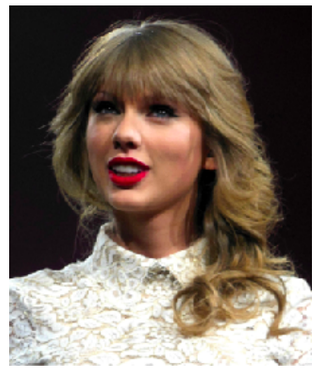


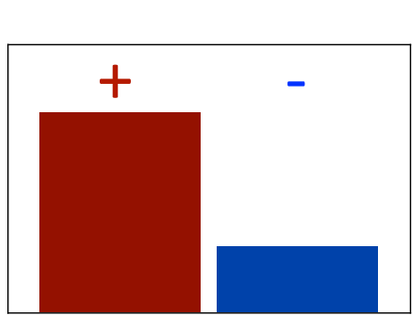
Personalized sentiment analysis

- **Goal**: personalized conditional likelihood, $p(y|\mathbf{x}, a)$.
  - $\mathbf{x}$ is the text, and a is the author.









- **Problem**: we have labeled examples for only a few authors.

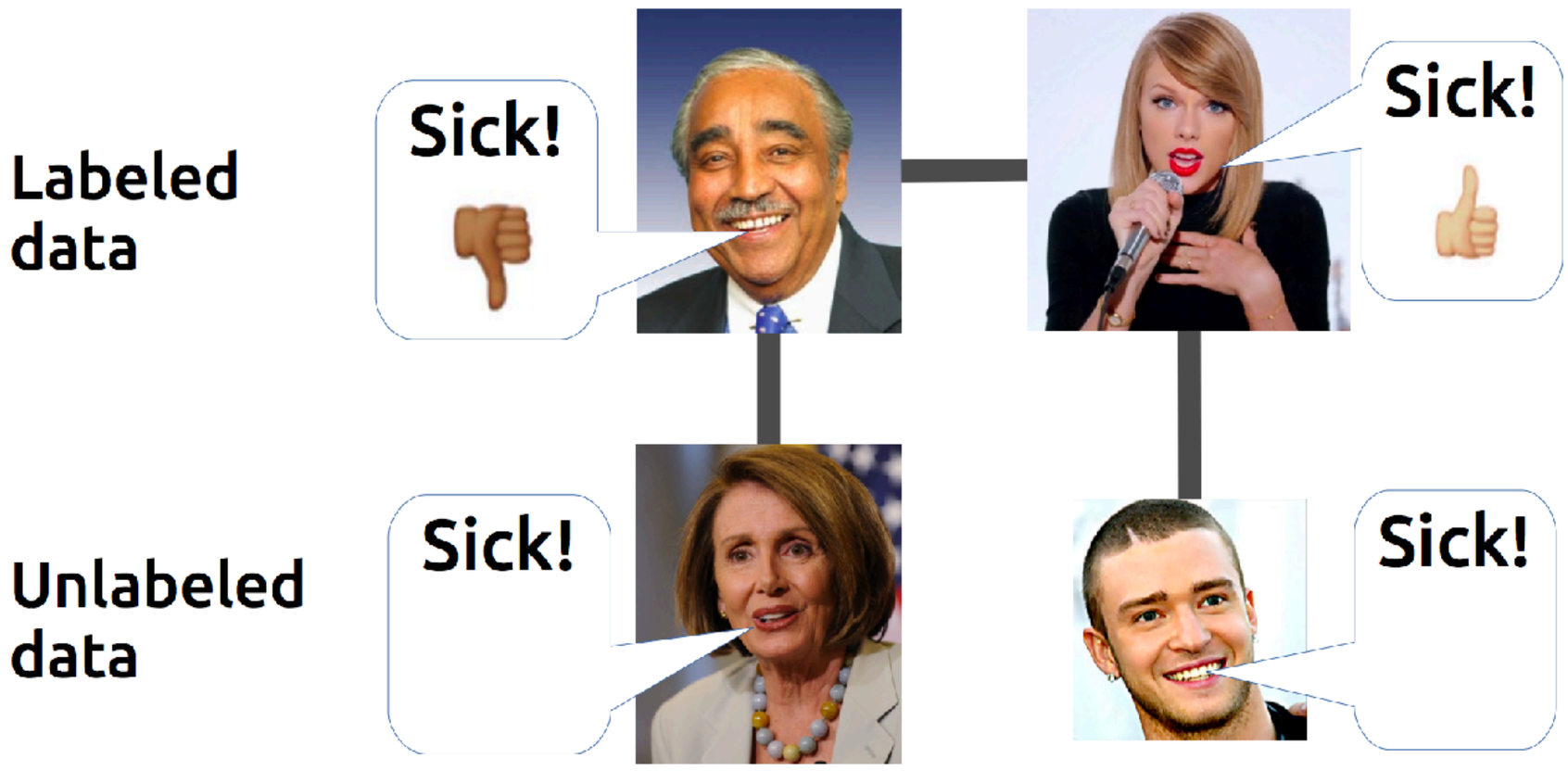
Homophily to the rescue?

Homophily: neighbors have similar properties.

Thelwall (2009); Al Zamal et al. (2012)

Homophily to the rescue?

Homophily: neighbors have similar properties.



Thelwall (2009); Al Zamal et al. (2012)

Evidence for linguistic homophily

Pilot study: is classifier accuracy assortative on the Twitter social network?

Evidence for linguistic homophily

Pilot study: is classifier accuracy assortative on the Twitter social network?

$$\text{assort}(G) = \frac{1}{\#|G|} \sum_{(i,j)\in G} \delta(y_i = \hat{y}_i)\delta(y_j = \hat{y}_j) + \delta(y_i \neq \hat{y}_i)\delta(y_j \neq \hat{y}_j)$$

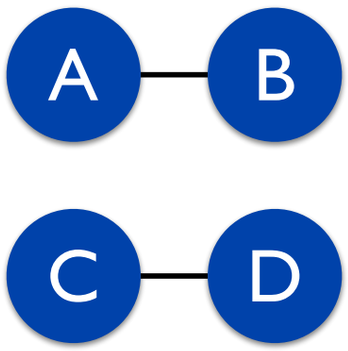
- Whether a sentiment classifier tends to make consistent predictions for social neighbors.

Evidence for linguistic homophily

Network rewiring: degree-preserving randomization

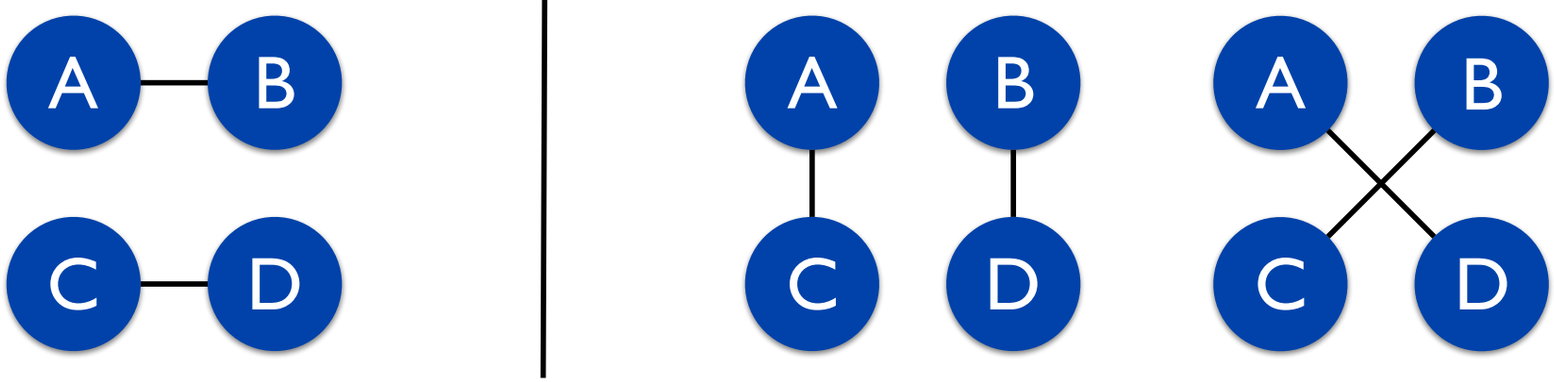
Evidence for linguistic homophily

Network rewiring: degree-preserving randomization



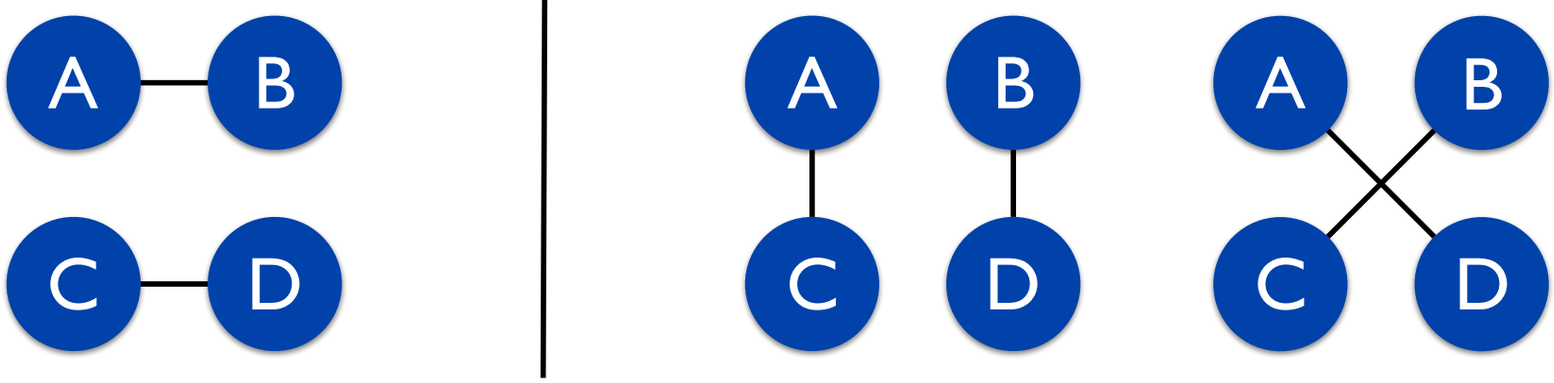
Evidence for linguistic homophily

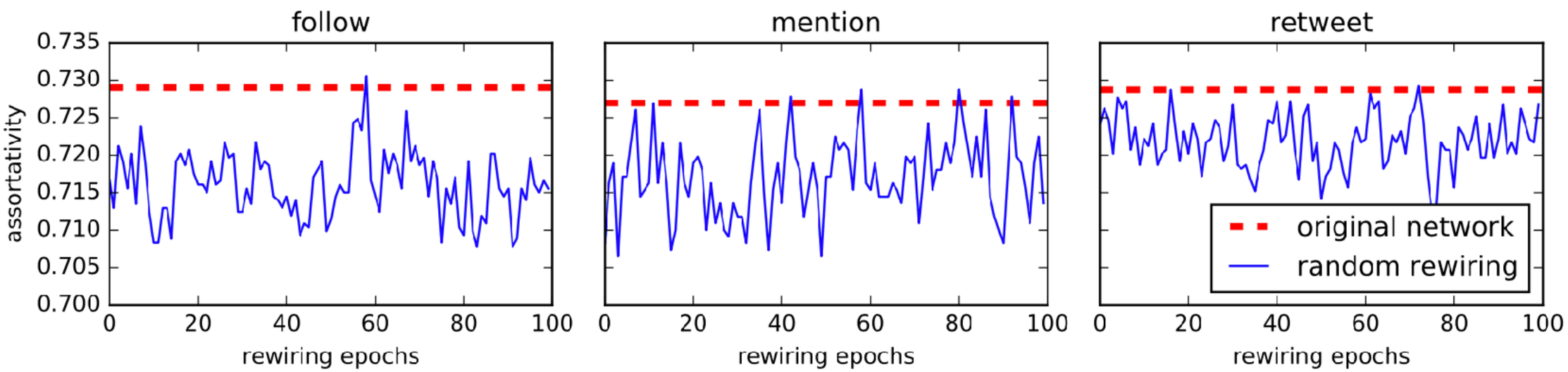
Network rewiring: degree-preserving randomization



Evidence for linguistic homophily

Network rewiring: degree-preserving randomization





Model

Personalization by ensemble

$$p(y|\mathbf{x}, a) = \sum_{k=1}^{K} \underbrace{Pr(Z_a = k|a, G)}_{\text{ensemble weights}} \times \underbrace{p(y|\mathbf{x}, Z_a = k)}_{\text{basis models}}$$

Personalization by ensemble

$$p(y|\mathbf{x}, a) = \sum_{k=1}^{K} \underbrace{Pr(Z_a = k|a, G)}_{\text{ensemble weights}} \times \underbrace{p(y|\mathbf{x}, Z_a = k)}_{\text{basis models}}$$

- Train each basis model with all the labeled data.
- Employ ConvNets as basis models.

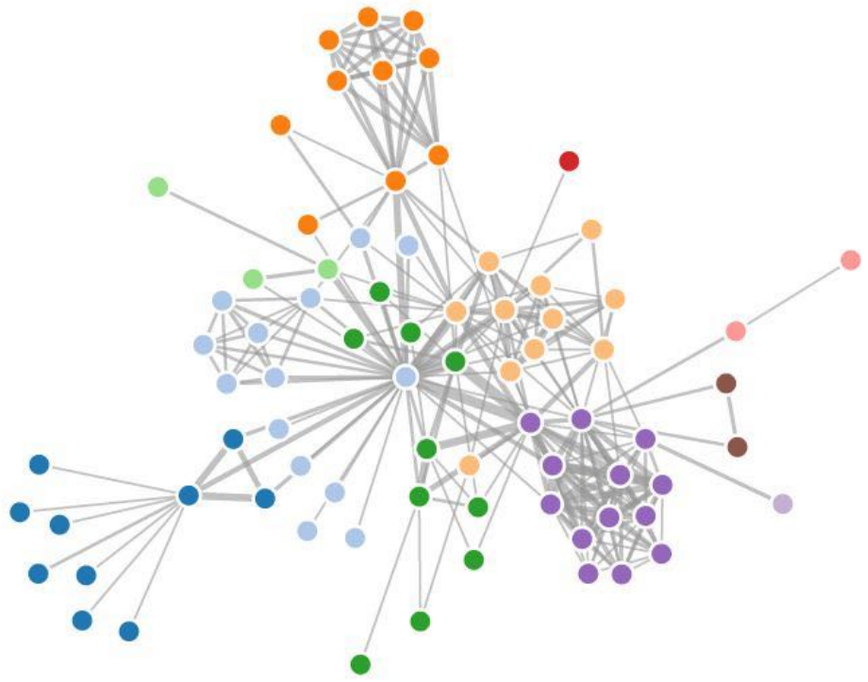
Personalization by ensemble

$$p(y|\mathbf{x}, a) = \sum_{k=1}^{K} \underbrace{Pr(Z_a = k|a, G)}_{\text{ensemble weights}} \times \underbrace{p(y|\mathbf{x}, Z_a = k)}_{\text{basis models}}$$

- Train each basis model with all the labeled data.
  - Employ ConvNets as basis models.
- Apply linguistic homophily:
  - Adopt similar ensemble weights for social neighbors.
  - De-correlate errors made by different basis models.

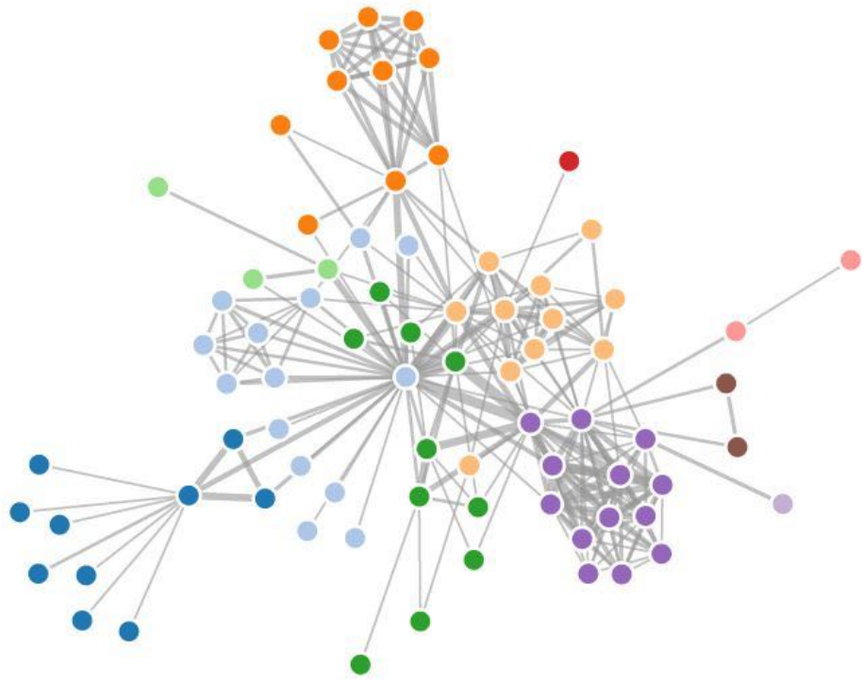
Network-driven personalization

- For each author, estimate a node embedding $\mathbf{v}_a$ (Tang et al., 2015).
- Nodes who share neighbors get similar embeddings.



Network-driven personalization

- For each author, estimate a node embedding $\mathbf{v}_a$ (Tang et al., 2015).
- Nodes who share neighbors get similar embeddings.



- **Social attention**:

$$Pr(Z_a = k|a, G) = \text{SoftMax}(f(\mathbf{v}_a))$$

Learning

- Jointly train with cross-entropy loss:

$$\ell(\Theta) = -\sum_{t=1}^{T} \mathbf{1}[Y^* = t] \log \Pr(Y = t \mid \mathbf{x}, a)$$

Learning

- Jointly train with cross-entropy loss:

$$\ell(\Theta) = -\sum_{t=1}^{T} \mathbf{1}[Y^* = t] \log \Pr(Y = t \mid \mathbf{x}, a)$$

Problem: network information tends to be ignored.

Learning

- Jointly train with cross-entropy loss:

$$\ell(\Theta) = -\sum_{t=1}^{T} \mathbf{1}[Y^* = t] \log \Pr(Y = t \mid \mathbf{x}, a)$$

Problem: network information tends to be ignored.

- Pre-train basis models with instance-weighted losses:

$$\ell_k = -\alpha_{a,k} \sum_{t=1}^{T} \mathbf{1}[Y^* = t] \log \Pr(Y = t \mid \mathbf{x}, Z_a = k)$$

Experiments

Data

- SemEval Twitter sentiment analysis data.
  - 18,024 tweets
  - Follow, mention, retweet networks
  - Network expansion

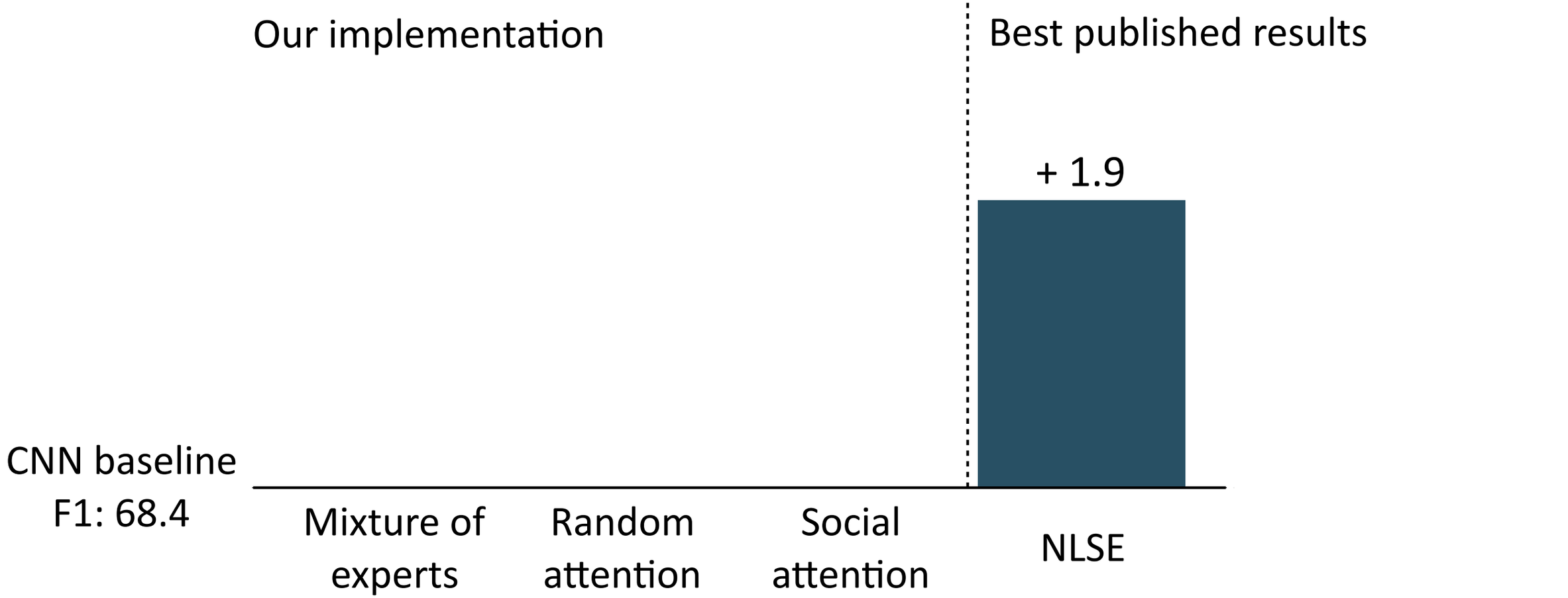
Nakov et al. (2013); Rosenthal et al. (2015); Tang et al. (2012)

Data

- SemEval Twitter sentiment analysis data.
  - 18,024 tweets
  - Follow, mention, retweet networks
  - Network expansion
- Ciao product review sentiment analysis data.
  - 100,000 reviews
  - User trust network

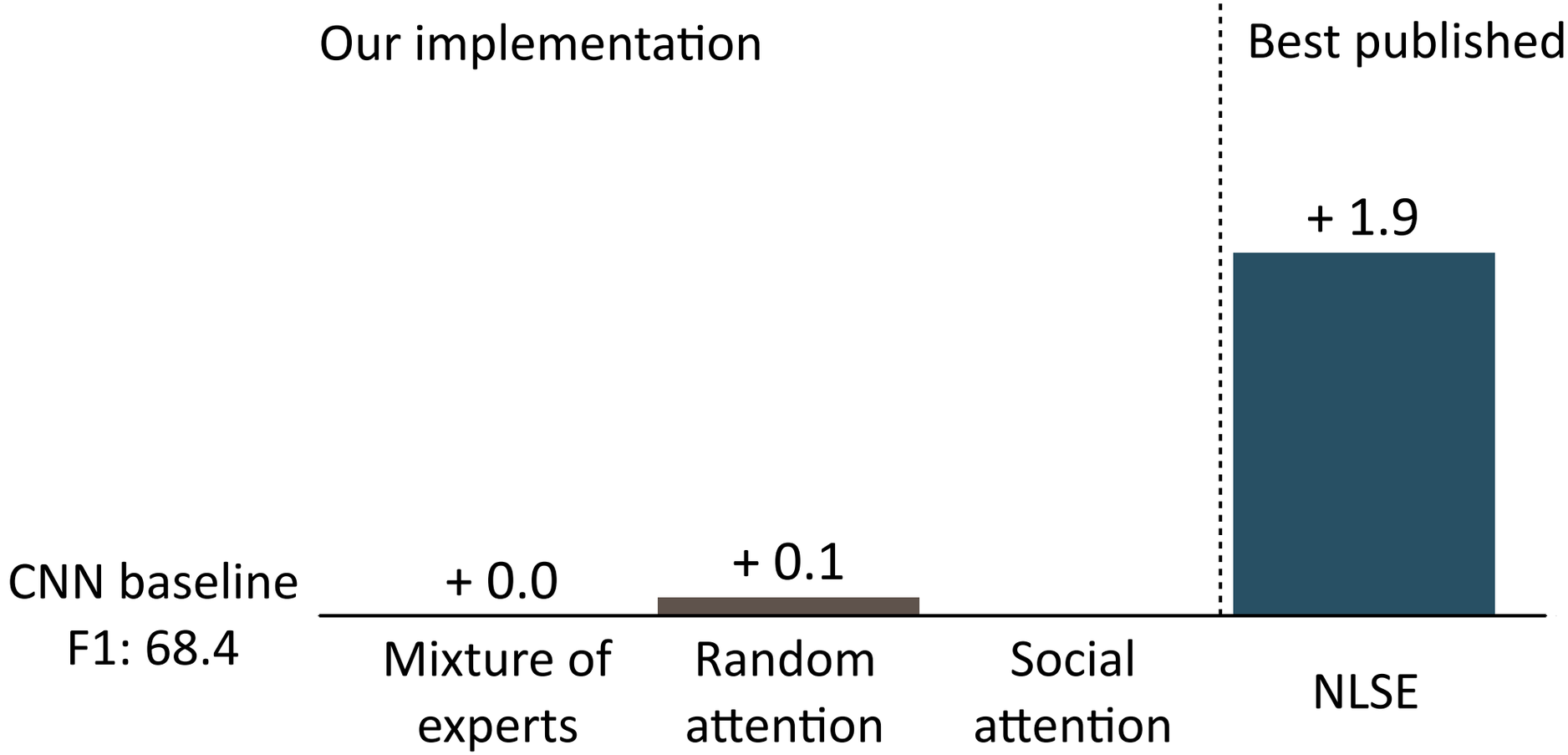
Nakov et al. (2013); Rosenthal et al. (2015); Tang et al. (2012)

Results: SemEval Twitter data



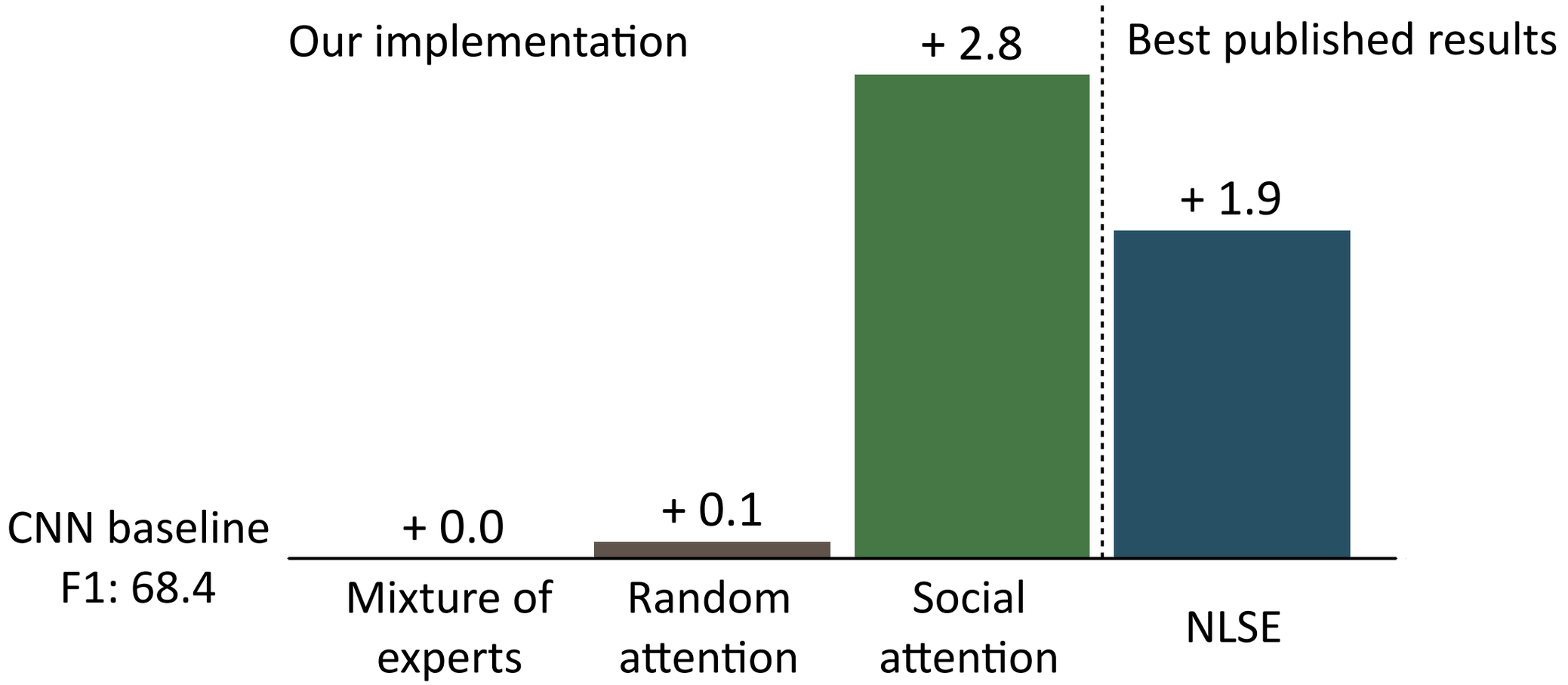
Astudillo et al. (2015)

Results: SemEval Twitter data



Astudillo et al. (2015)

Results: SemEval Twitter data



Astudillo et al. (2015)

Variable sentiment words

	More positive	More negative
1	banging loss fever broken fucking	dear like god yeah wow
2	chilling cold ill sick suck	satisfy trust wealth strong lmao
3	ass damn piss bitch shit	talent honestly voting win clever
4	insane bawling fever weird cry	lmao super lol haha hahaha
5	ruin silly bad boring dreadful	*lovatics* wish *beliebers arianators kendall*

Results: Ciao review data

CNN baseline
F1: 74.4

+ 0.0

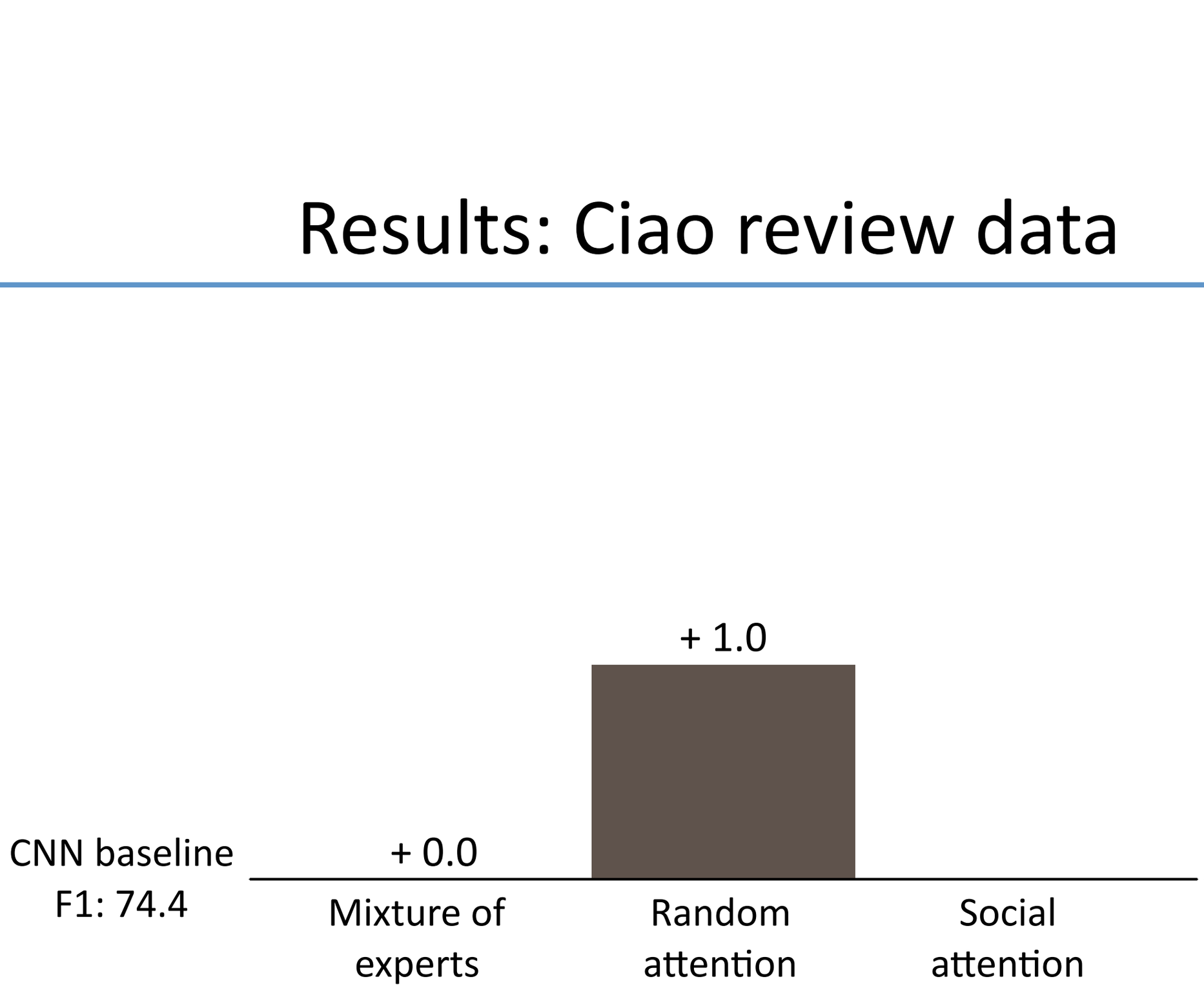
Mixture of experts

Random attention

Social attention

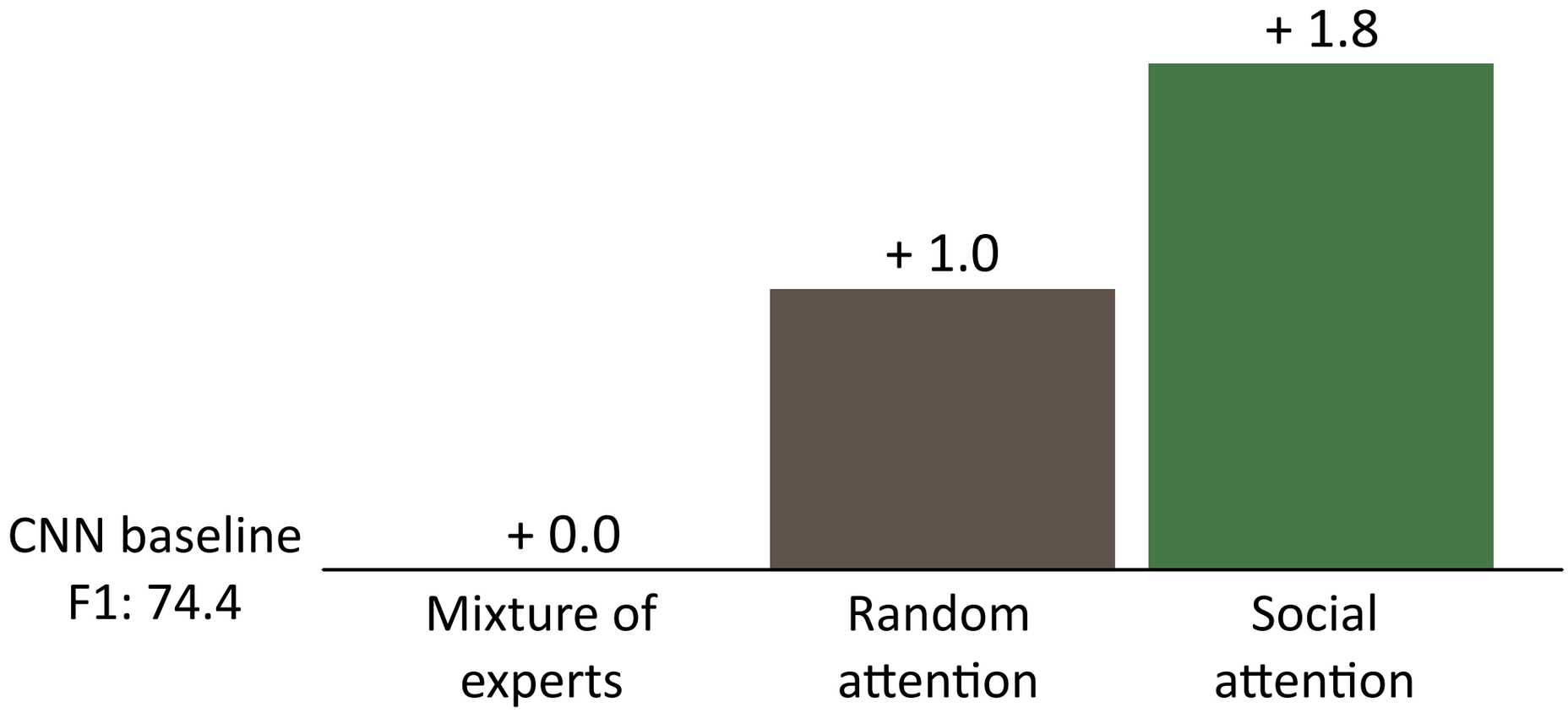
Tang et al. (2012)

Results: Ciao review data



Tang et al. (2012)

Results: Ciao review data



Tang et al. (2012)

Conclusions and future work

- Language variation poses challenges in sentiment analysis.
- Linguistic homopily alleviates the data sparsity issue for estimating personalized models.
- Social attention mechanism significantly improves accuracy.
- The socially-infused ensemble architecture can be applied to other tasks such as tagging, parsing, etc.

Follow @cocoweixu

Instructor: Wei Xu

www.cis.upenn.edu/~xwe/

Course Website: socialmedia-class.org