(Deep) Neural Network & Text Mining

Piji Li

lipiji.pz@gmail.com

香港中文大學
The Chinese University of Hong Kong

Outline

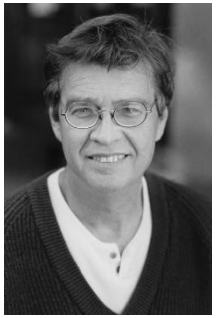
- Deep Learning
  - Story
- DL for NLP & Text Mining
  - Words
  - Sentences
  - Documents

Outline

- Deep Learning
  - Story
- DL for NLP & Text Mining
  - Words
  - Sentences
  - Documents

BP

1986



D.E. Rumelhart, G.E. Hinton, R.J. Williams
Learning representation by back-propagating errors. *Nature*, 323 (1986), pp. 533–536

- Solved learning problem
- Biological system
- …

- Hard to train (non-convex, tricks)
- Hard to do theoretical analysis
- Small training sets ...

DBN

2006

Hinton, Geoffrey, Simon Osindero, and Yee-Whye Teh. "**A fast learning algorithm for deep belief nets.**" *Neural computation* 18, no. 7 (2006): 1527-1554.
Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "**Reducing the dimensionality of data with neural networks**." *Science* 313, no. 5786 (2006): 504-507.

- Unsupervised layer-wised pre-training
- Supervised fine-tuning
- Feature learning & distributed representations
- Large scale datasets
- Tricks (learning rate, mini-batch size, momentum, etc.)

- https://github.com/lipiji/PG_DEEP

CD-DNN-HMM

2011

PhD candidate of Prof. Hinton, intern at MSR

Dahl, George E., Dong Yu, Li Deng, and Alex Acero. "**Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition**." *Audio, Speech, and Language Processing, IEEE Transactions on* 20, no. 1 (2012): 30-42.

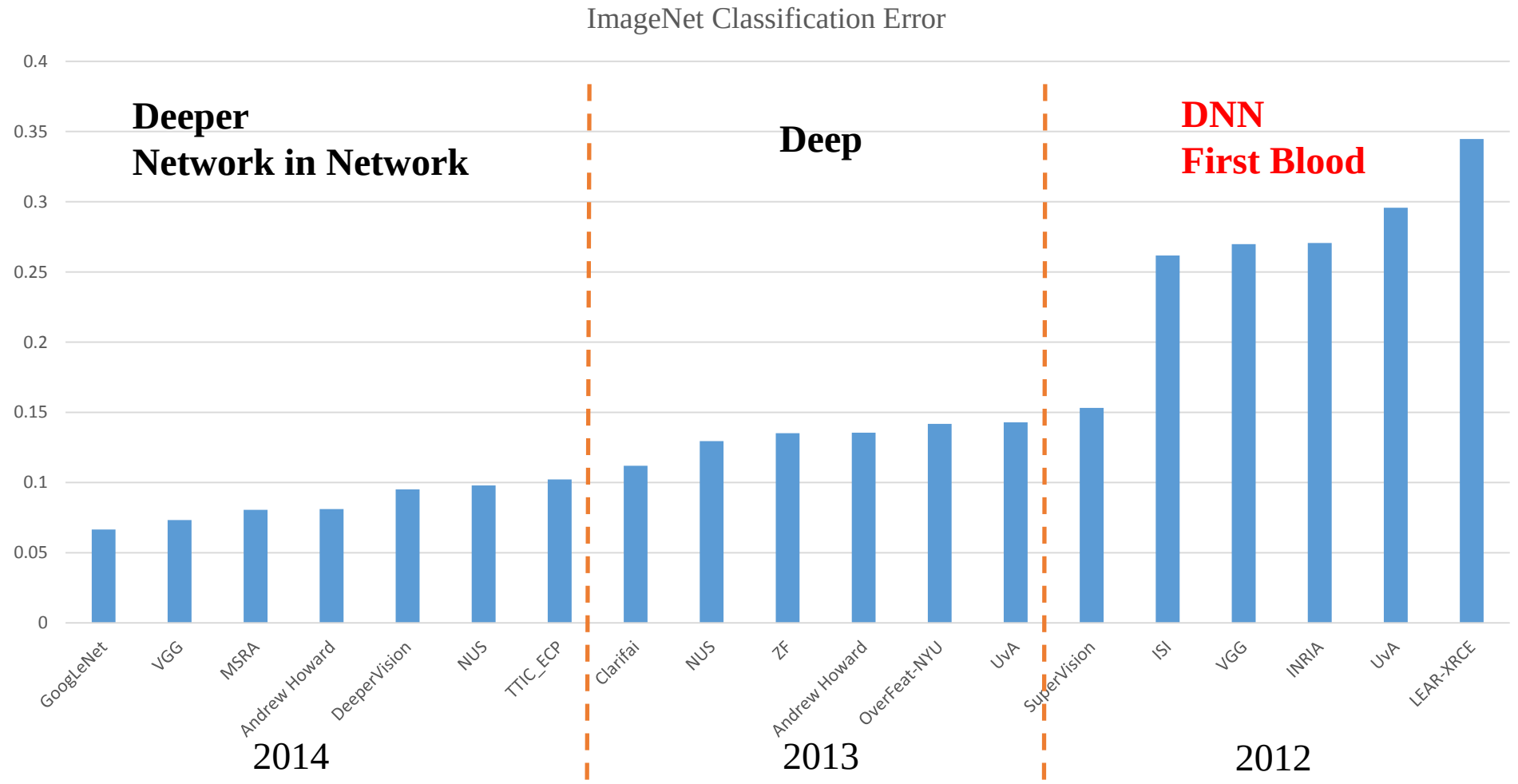
Winner of the 2013 IEEE Signal Processing Society Best Paper Award

GPU

task	hours of training data	DNN-HMM	GMM-HMM with same data
Switchboard (test set 1)	309	18.5	27.4
Switchboard (test set 2)	309	16.1	23.6
English Broadcast News	50	17.5	18.8
Bing Voice Search (Sentence error rates)	24	30.4	36.2
Google Voice Input	5,870	12.3	
Youtube	1,400	47.6	52.3

ImageNet Classification

- **1000** categories and **1.2** million training images



Li Fei-Fei: ImageNet Large Scale Visual Recognition Challenge, 2014 http://image-net.org/

Nowadays




People: Geoffrey Hinton, Yann LeCun, Yoshua Bengio, Andrew Ng
Industry: Google, Baidu, Facebook, IBM
Paper: ICCV CVPR ECCV; ICML, NIPS, AAAI, ICLR; ACL, EMNLP, COLING

Startup Companies using (Deep) Machine Learning:

DEEPMIND clarifai MADBITS FACE++ CURALATE BINATIX Machine Intelligence

eMOTIENT cortica In Every Image PredictionIO vicarious SWIFT IQ EUVISION technologies

ReKognition AlchemyAPI™ ersatz labs SKYMIND 博昂科技 BOON INFO TECH LTD. DEEPGLINT 格灵深瞳

DNNresearch 
VISENZE INTELLIGENT VISUAL RECOGNITION 
Graymatics Ultimate Multimedia Sensing 

Outline

- Deep Learning
  - Story
- DL for NLP & Text Mining
  - Words
  - Sentences
  - Documents

NN & Words

- Distributed Representation
  - A Neural Probabilistic Language Model
  - Word2Vec
- Application
  - Semantic Hierarchies of Words
  - Machine Translation

$$V(\text{king}) - V(\text{queen}) + V(\text{woman}) \approx V(\text{man})$$

Neural Probabilistic Language Model

- Bengio, Yoshua, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. "**A Neural Probabilistic Language Model**." *Journal of Machine Learning Research* 3 (2003): 1137-1155.

  - Maximizes the penalized log-likelihood:

$$L = \frac{1}{T}\sum_t \log f(w_t, w_{t-1}, \cdots, w_{t-n+1}; \theta) + R(\theta)$$

  - Sofmax

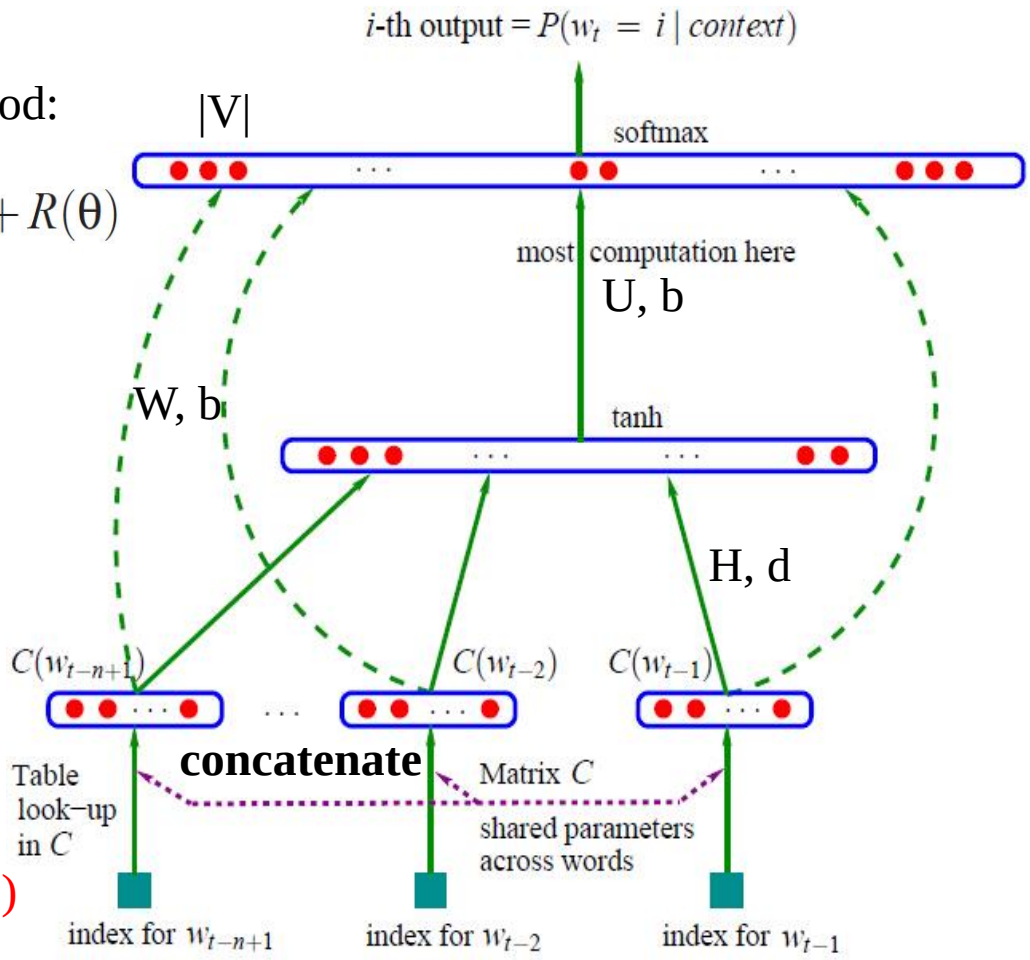
Time-Consuming

$$\hat{P}(w_t | w_{t-1}, \cdots w_{t-n+1}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}}$$

  - where

$$y = b + Wx + U\tanh(d + Hx)$$

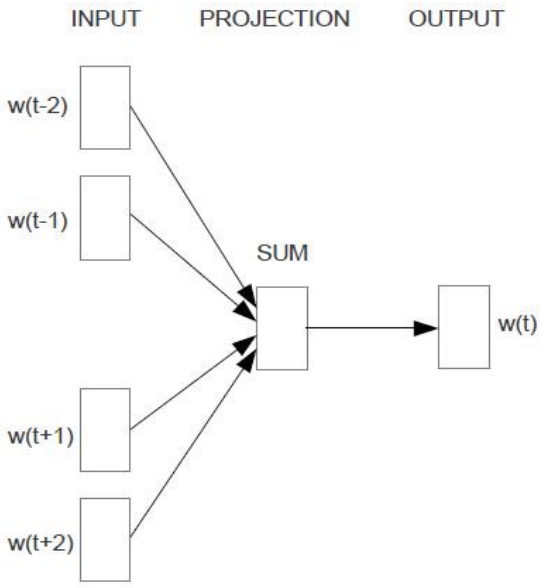
  - Training: BP



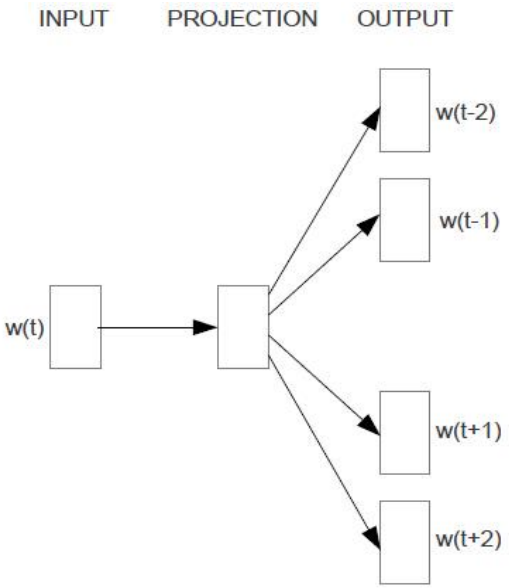
Better than n-gram model
1, Word vector;
2, Need not soothing, p(w|context) ~(0,1)

Word2Vec

- Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "**Efficient estimation of word representations in vector space**." *ICLR* (2013).
  - Large improvements in accuracy, lower computational cost.
  - It takes less than a day to train from 1.6 billion words data set.



CBOW



Skip-gram

$$\mathcal{L} = \sum_{w \in \mathcal{C}} \log p(w|Context(w))$$

$$\mathcal{L} = \sum_{w \in \mathcal{C}} \log p(Context(w)|w)$$

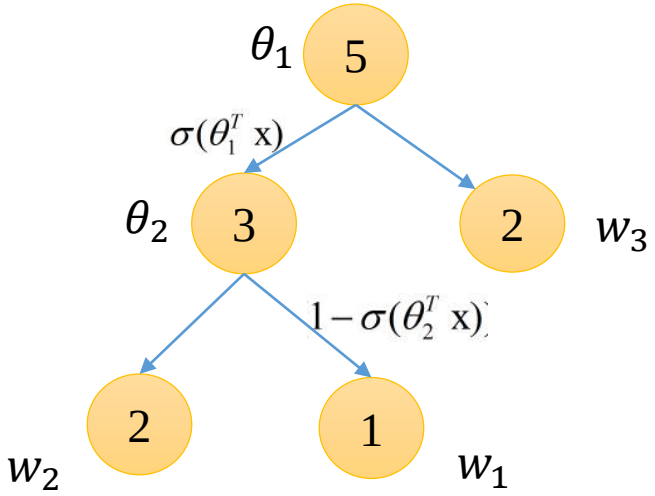
Word2Vec - Hierarchical Softmax

- Morin, Frederic, and Yoshua Bengio. "**Hierarchical probabilistic neural network language model**." In *AISTATS*, vol. 5, pp. 246-252. 2005.
- Mnih, Andriy, and Geoffrey E. Hinton. "**A scalable hierarchical distributed language model**." In *Advances in neural information processing systems*, pp. 1081-1088. 2009.

Time-Consuming

$$\hat{P}(w_t|w_{t-1},\cdots w_{t-n+1}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}}$$

$$y = b + Wx + U\tanh(d + Hx)$$



- Hierarchical Strategies:
  O(|V|)→O(log|V|)

- **Word2Vec**:
  Huffman Tree - short codes to frequent words

$$p(\mathrm{w}_1 \mid \mathrm{C}_1) = \sigma(\theta_1^T \mathrm{x}) \times (1 - \sigma(\theta_2^T \mathrm{x}))$$

Word2vec - CBOW

- **Log-likelihood**

$$\mathcal{L} = \sum_{w \in \mathcal{C}} \log p(w|Context(w))$$

- **Condition Probability**

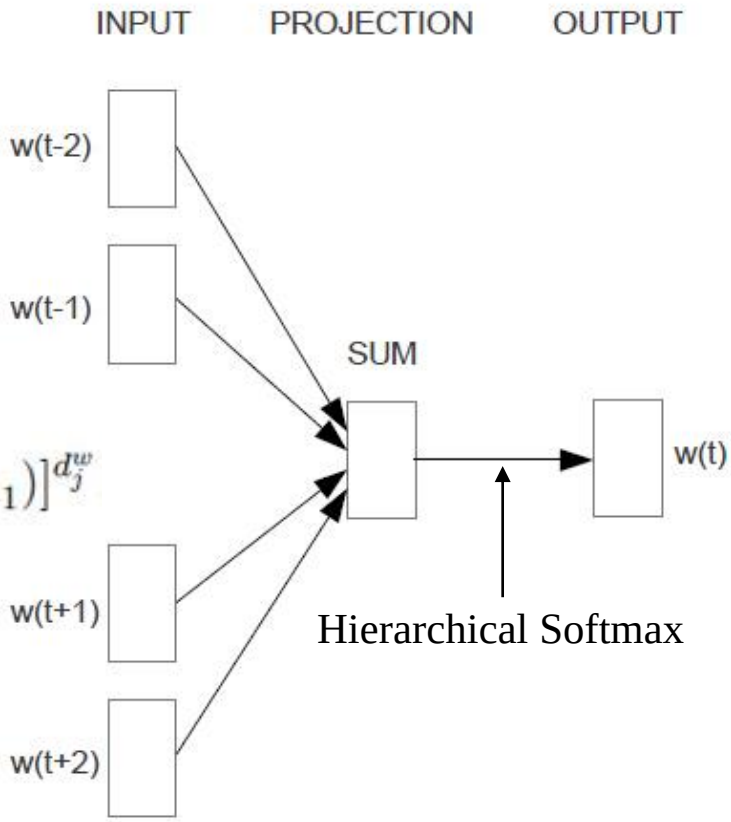
$$p(w \mid Context(w)) = \prod_{j=2}^{l^w} p(d_j^w \mid x_w, \theta_{j-1}^w)$$

$$p(d_j^w|\mathbf{x}_w, \theta_{j-1}^w) = [\sigma(\mathbf{x}_w^\top \theta_{j-1}^w)]^{1-d_j^w} \cdot [1 - \sigma(\mathbf{x}_w^\top \theta_{j-1}^w)]^{d_j^w}$$

- **SGD**

$$\theta_{j-1}^w := \theta_{j-1}^w + \eta[1 - d_j^w - \sigma(x_w^T \theta_{j-1}^w)]x_w$$

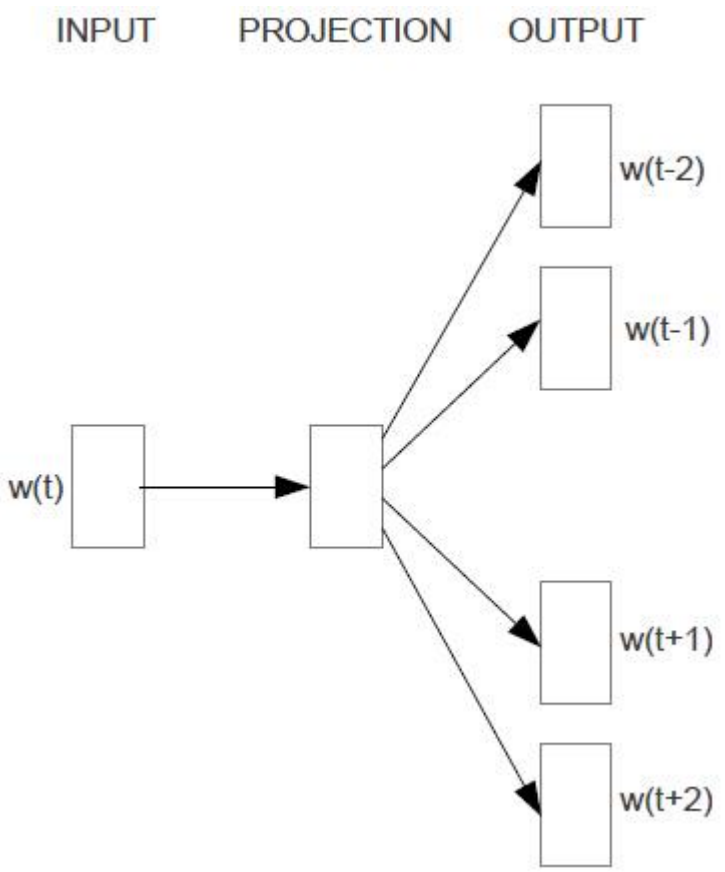
$$\mathbf{v}(\tilde{w}) := \mathbf{v}(\tilde{w}) + \eta \sum_{j=2}^{l^w} \frac{\partial \mathcal{L}(w,j)}{\partial \mathbf{x}_w}, \quad \tilde{w} \in Context(w)$$



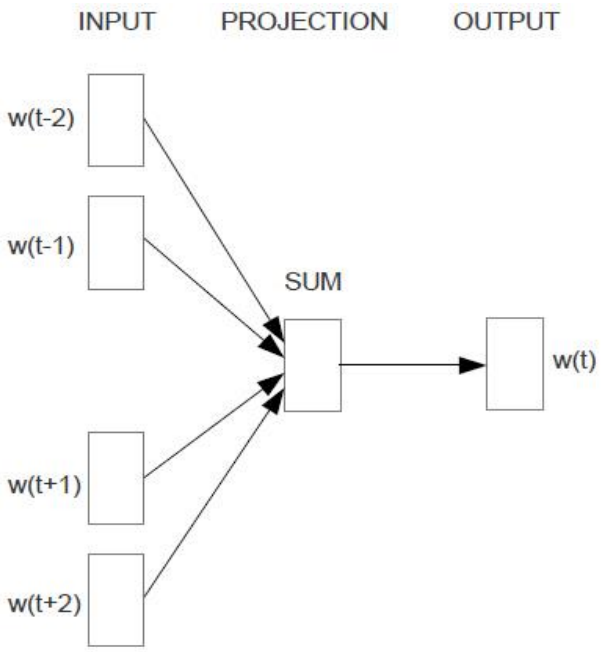
Blog: http://blog.csdn.net/itplus/article/details/37969979

Word2vec – Skip-gram

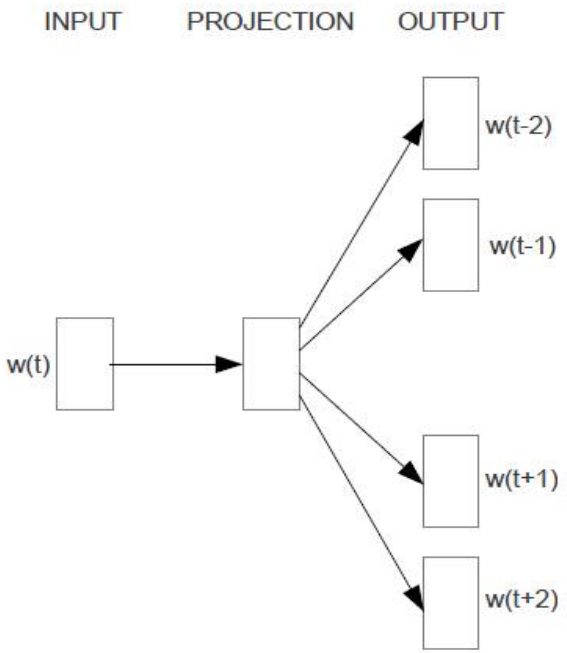
$$p(Context(w)|w) = \prod_{u \in Context(w)} p(u|w)$$



CBOW vs Skip-gram



CBOW



Skip-gram

Syntactic Relation

Semantic Relation

Word2Vec - Negative Sampling

Mikolov, Tomas, et al. "**Distributed representations of words and phrases and their compositionality**." *NIPS* (2013).

$$g(w) = \prod_{u \in \{w\} \cup NEG(w)} p(u|Context(w))$$

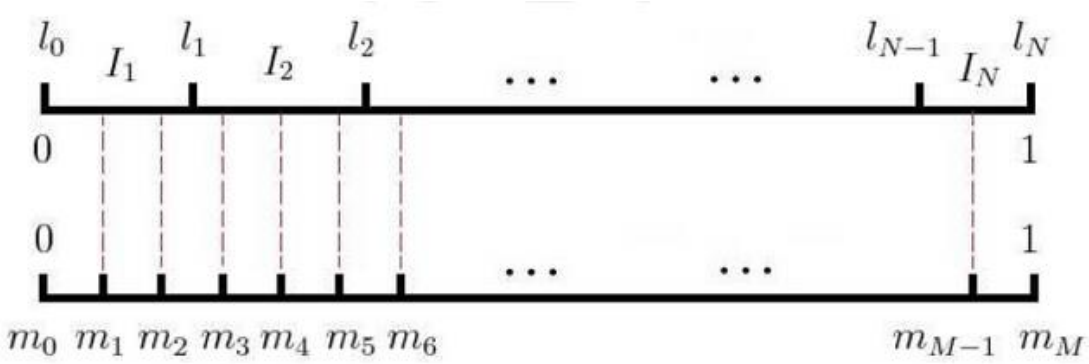
Phrase Skip-Gram Results

$$\text{score}(w_i, w_j) = \frac{\text{count}(w_i w_j) - \delta}{\text{count}(w_i) \times \text{count}(w_j)}$$

Subsampling:

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}}$$

Negative Sampling:



CBOW & Skip-gram

Blog: http://blog.csdn.net/itplus/article/details/37969979

Word2Vec Properties

- Vector spaces
  - Mapping different vector spaces
    - Machine Translation
    - …
- Embedding offsets
  - V(king) – V(queen) + V(woman) ≈ V(man)
    - Complicated Semantic Relations: Hierarchies
    - …
- ?

Word2Vec Properties

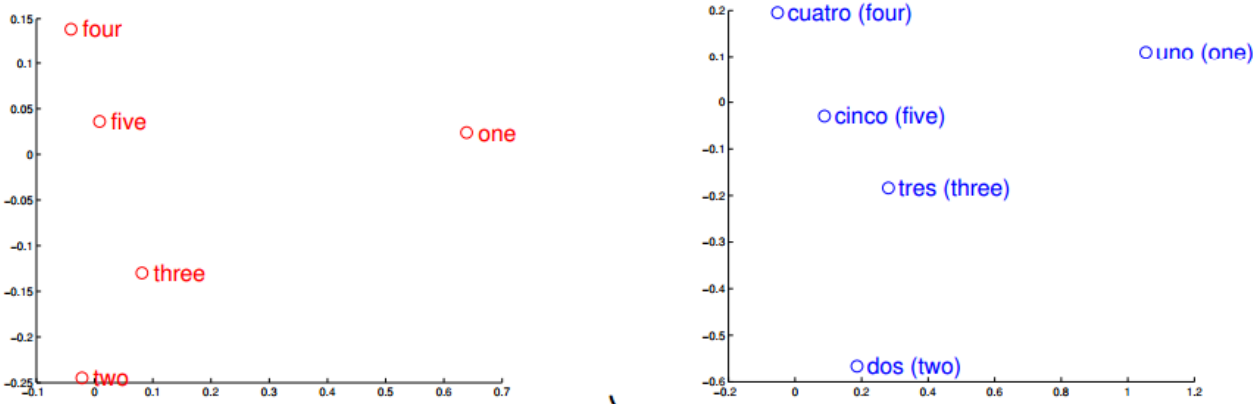
- Vector spaces
  - Mapping different vector spaces
    - Machine Translation
    - …
- Embedding offsets
  - V(king) – V(queen) + V(woman) ≈ V(man)
    - Complicated Semantic Relations: Hierarchies
    - …
- ?

Machine Translation

- Mikolov, Tomas, Quoc V. Le, and Ilya Sutskever. "**Exploiting similarities among languages for machine translation**." *arXiv preprint arXiv:1309.4168* (2013).

Translation Matrix

$$\min_W \sum_{i=1}^{n} \|Wx_i - z_i\|^2$$



English (left) → Spanish (right).

- Devlin, Jacob, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. "**Fast and Robust Neural Network Joint Models for Statistical Machine Translation**." **ACL 2014 Best Long Paper**.
- Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. "**Neural Machine Translation by Jointly Learning to Align and Translate**." *arXiv preprint arXiv:1409.0473* (2014).

Word2Vec Properties

- Vector spaces
  - Mapping different vector spaces
    - Machine Translation
    - …
- Embedding offsets
  - V(king) – V(queen) + V(woman) ≈ V(man)
    - Complicated Semantic Relations: Hierarchies
    - …
- ?

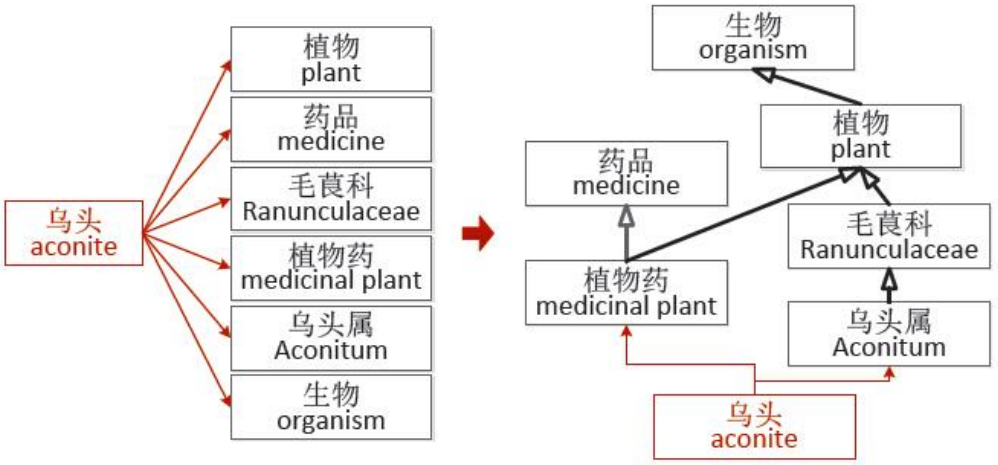
Semantic Hierarchies

- Fu, Ruiji, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. "**Learning semantic hierarchies via word embeddings**." ACL, 2014.

$$v(\texttt{king}) - v(\texttt{queen}) \approx v(\texttt{man}) - v(\texttt{woman})$$

Clustering **y - x**, and then for each cluster:

$$\Phi_k^* = \arg\min_{\Phi_k} \frac{1}{N_k} \sum_{(x,y)\in C_k} \| \Phi_k x - y \|^2$$



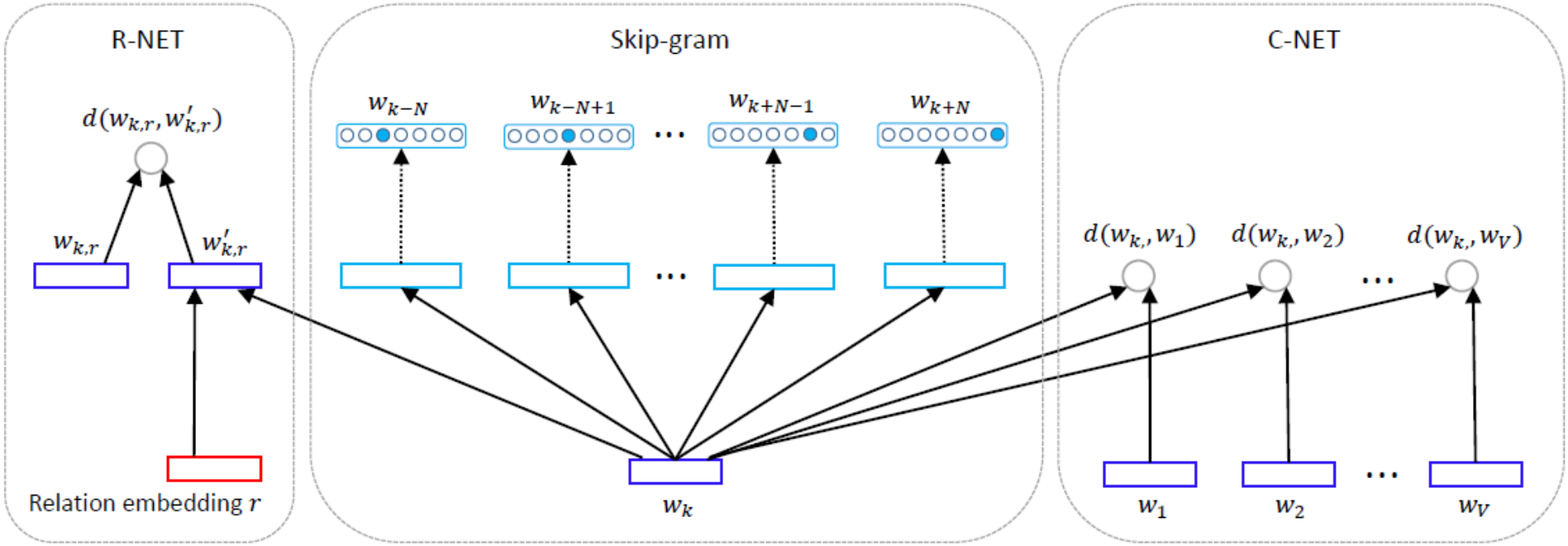
hypernym – hyponym

More Relations?

Figure 1: An example of semantic hierarchy construction.

RC-NET

- Xu, Chang, Yalong Bai, Jiang Bian, Bin Gao, Gang Wang, Xiaoguang Liu, and Tie-Yan Liu. "**RC-NET: A General Framework for Incorporating Knowledge into Word Representations**." CIKM (2014).

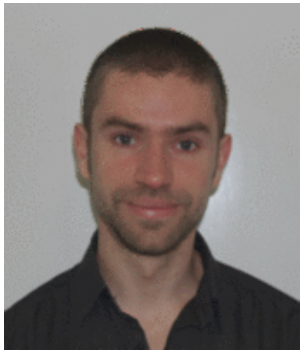


$$J = \alpha E_r + \beta E_c - L$$

$$E_r = \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'_{(h,r,t)}} \left[\gamma + d(h+r,t) - d(h'+r,t')\right] \quad E_c = \sum_{i=1}^{V}\sum_{j=1}^{V} s(w_i, w_j) d(w_i, w_j)$$

NN & Sentences

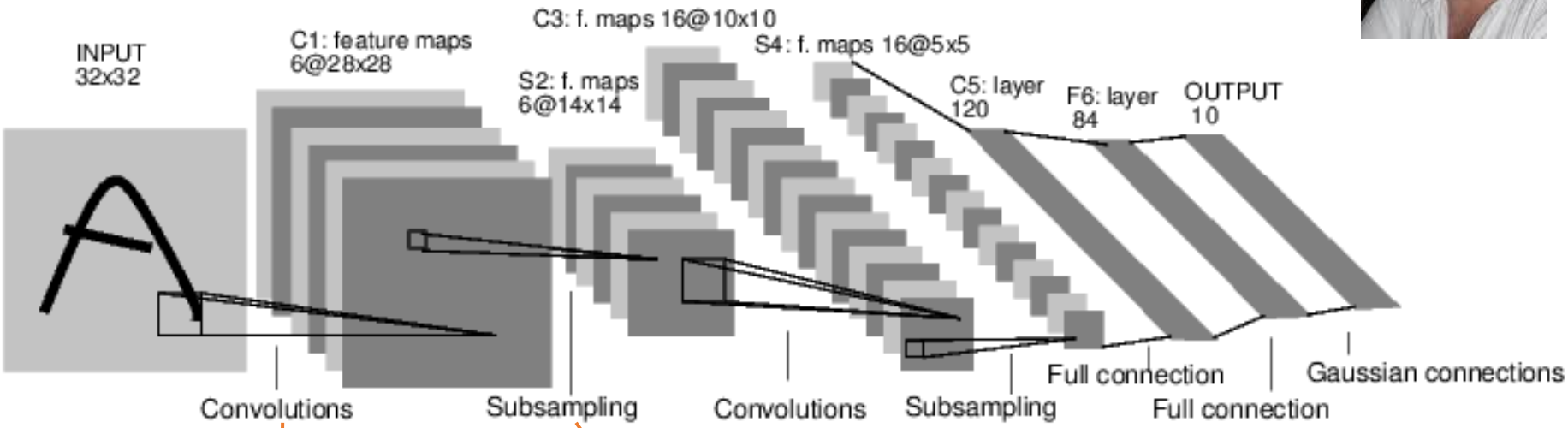
- Blunsom, Phil, Edward Grefenstette, and Nal Kalchbrenner. "**A Convolutional Neural Network for Modelling Sentences**." ACL 2014.
- Hermann, Karl Moritz, and Phil Blunsom. "**Multilingual Models for Compositional Distributed Semantics**." *arXiv preprint arXiv:1404.4641* (2014).
- Hermann, Karl Moritz, and Phil Blunsom. "**Multilingual Distributed Representations without Word Alignment**." *arXiv preprint arXiv: 1312.6173*(2013).
- Kim, Yoon. "**Convolutional Neural Networks for Sentence Classification**. " arxiv : 2014
- Le, Quoc V., and Tomas Mikolov. "**Distributed Representations of Sentences and Documents.**" *ICML* (2014).
- Baotian Hu, Zhengdong Lu, Hang Li, etc. "**Convolutional Neural Network Architectures for Matching Natural Language Sentences**." NIPS 2014



Phil Blunsom
Oxford
http://www.cs.ox.ac.uk/people/phil.blunsom/

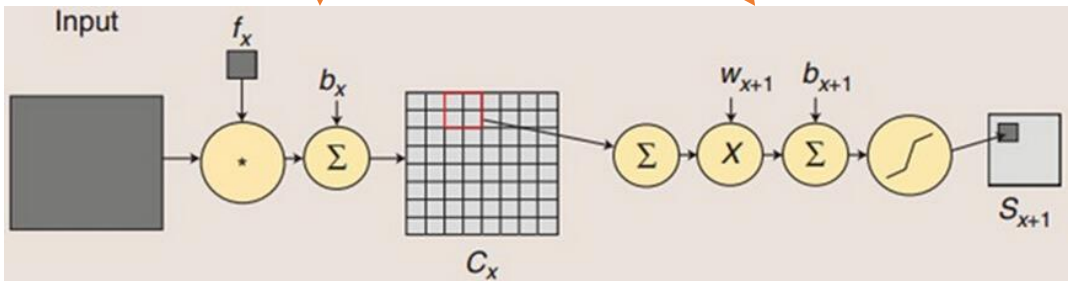
Convolutional Neural Networks

- Best Results in Computer Vision (LeNet5, ImageNet 2012~now)
- Shared Weight: Convolutional Filter → Feature Map
- * Invariance: pooling (mean-, max-, stochastic)



C1: (5*5+1)*6=156 parameters;
156*(28*28)=122,304 connections

http://yann.lecun.com/exdb/lenet/



Convolutional Windows in Sentences

Relation Classification

- Zeng, Daojian, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. "**Relation Classification via Convolutional Deep Neural Network**." **COLING 2014 Best Paper**

  Cause-Effect relationship: "The [fire]e1 inside WTC was caused by exploding [fuel]e2"

SENNA

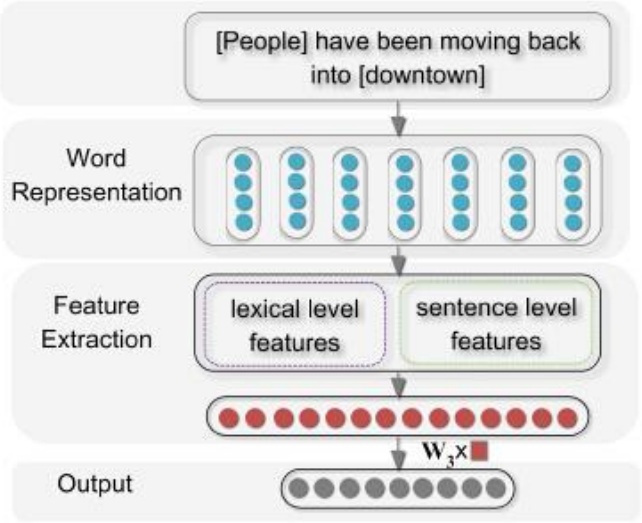


Figure 1: Architecture of the neural network used for relation classification.

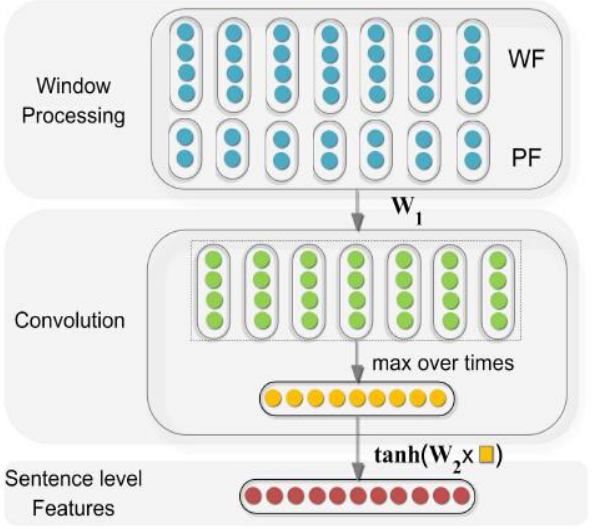


Figure 2: The framework used for extracting sentence level features.

Steps: 1, word vector; 2, lexical level features; 3, **sentences features via CNN**; 4, Concatenate features and feed into softmax classifier

CNN for Modelling Sentences

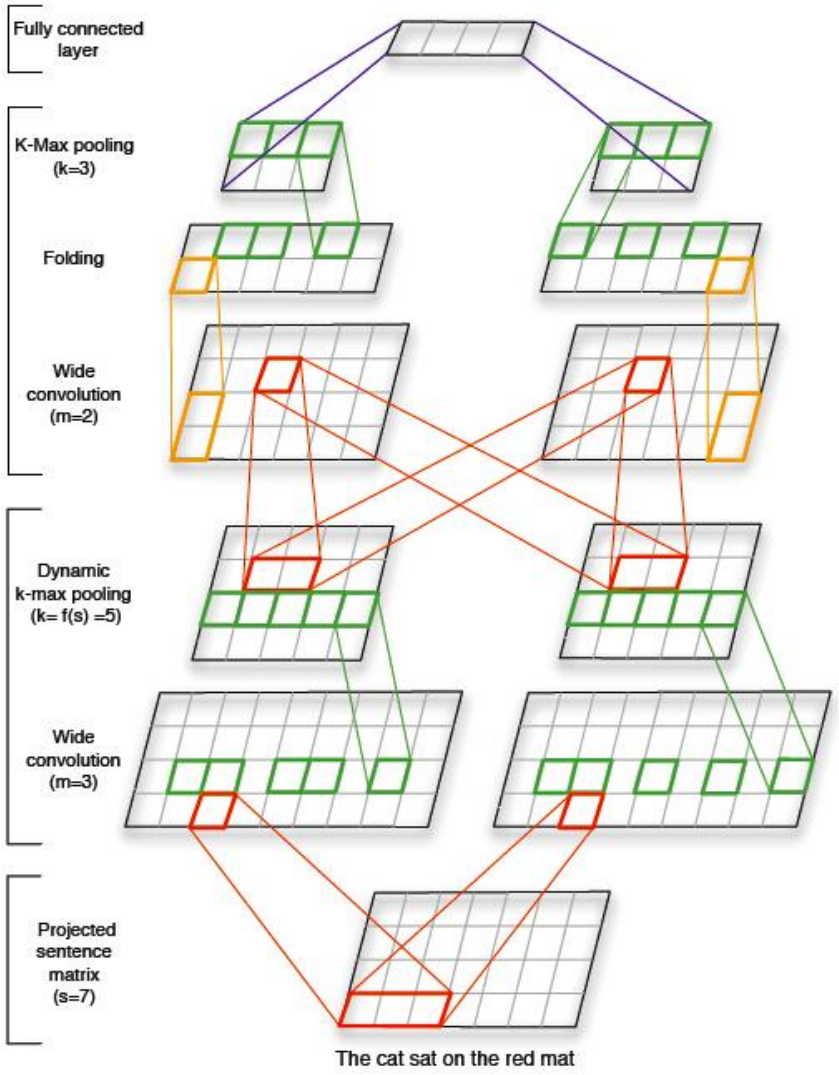
- Blunsom, Phil, Edward Grefenstette, and Nal Kalchbrenner. "**A Convolutional Neural Network for Modelling Sentences**." ACL 2014.

  - Word vector: learning
  - Supervisor depends on tasks
  - **Dynamic k-Max Pooling**

$$k_l = \max(\, k_{top},\, \lceil \frac{L-l}{L} s \rceil \,)$$

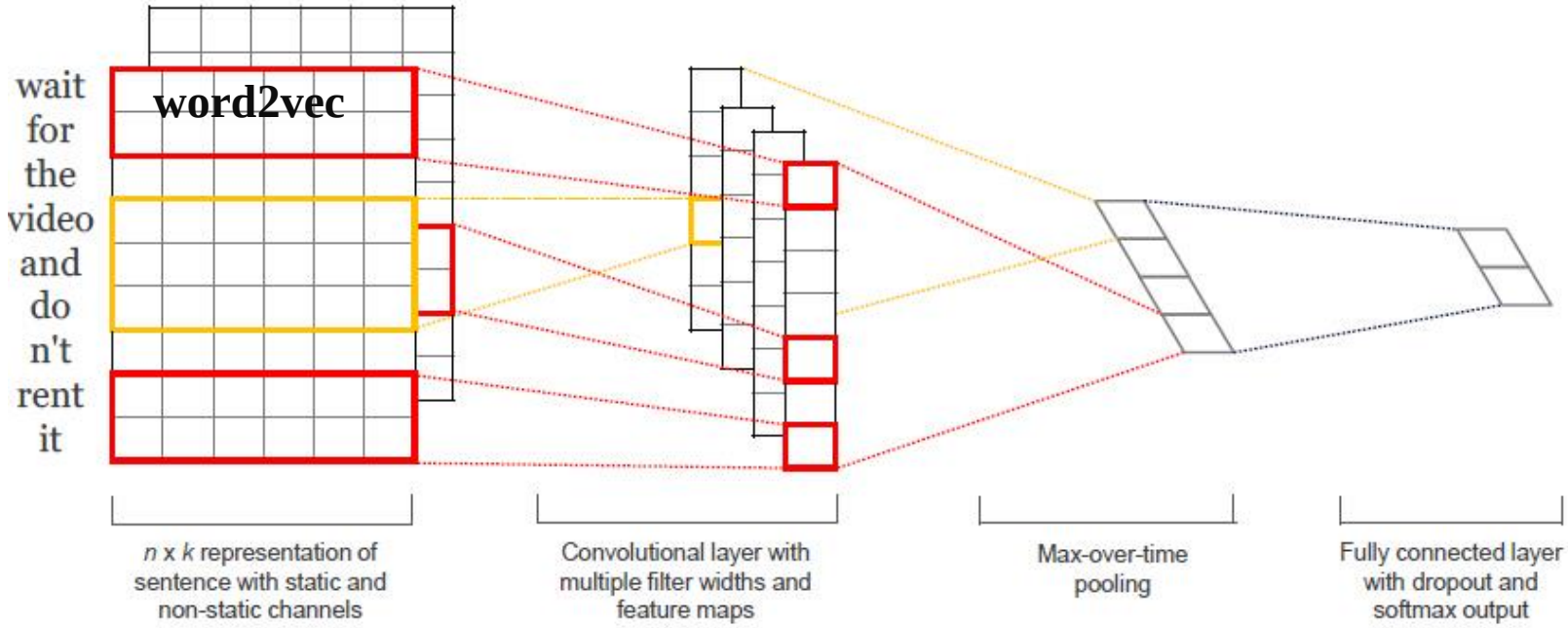
  - Folding
    - Sums every two rows
  - Multiple Feature Maps
    - Different features

- Word order, n-gram
- Sentence Feature
- Many tasks



CNN for Sentence Classification

- Kim, Yoon. "**Convolutional Neural Networks for Sentence Classification**. " arxiv : 2014



New: Multi-Channel

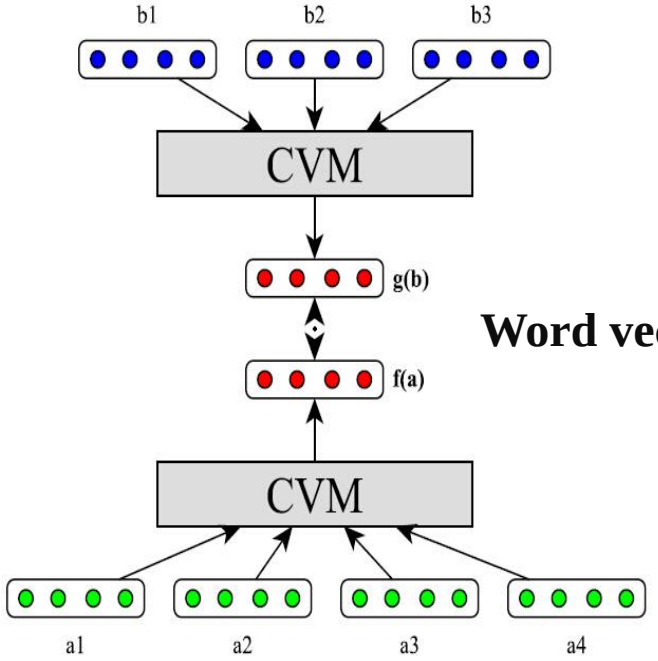
A model with two sets of word vectors. Each set of vectors is treated as a 'channel' and each filter is applied to both channels.

CNN for Matching

- Baotian Hu, Zhengdong Lu, Hang Li, etc. "**Convolutional Neural Network Architectures for Matching Natural Language Sentences**." NIPS 2014
  - No pdf yet

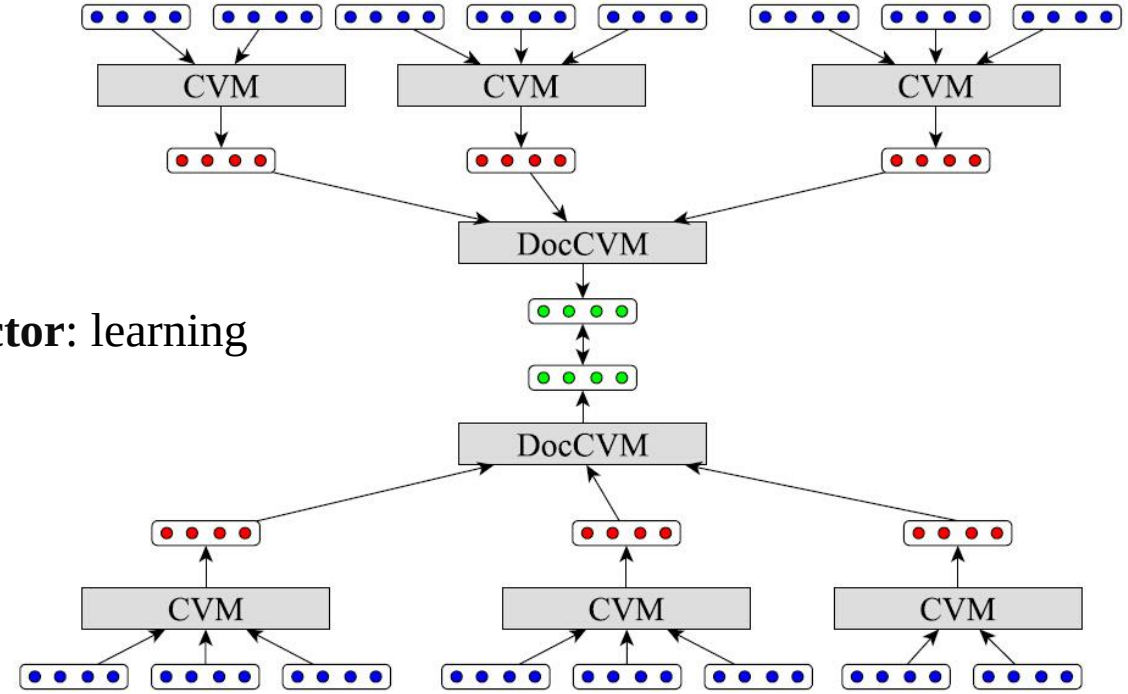
Multilingual Models

- Hermann, Karl Moritz, and Phil Blunsom. "**Multilingual Models for Compositional Distributed Semantics**." *arXiv preprint arXiv:1404.4641* (2014).



Sentences

Word vector: learning



Documents

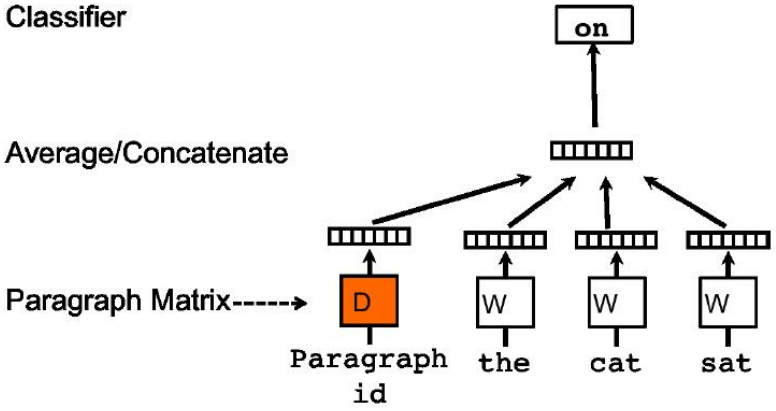
Objection:

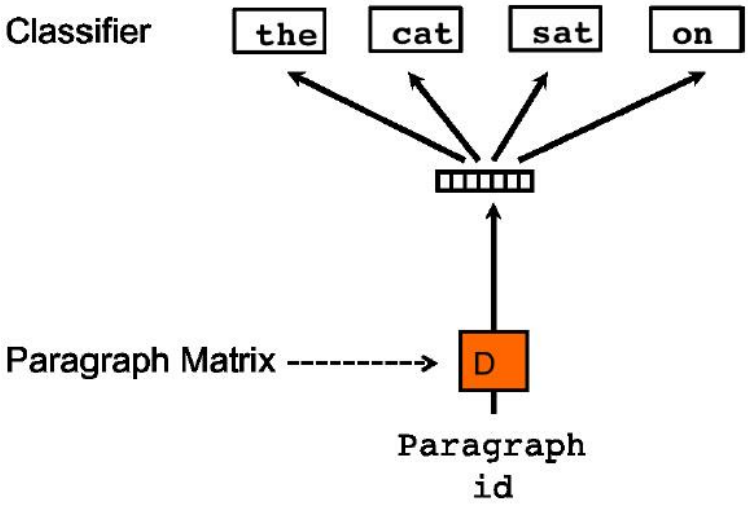
$$E_{bi}(a,b) = \|f(a) - g(b)\|^2$$

CVM: SUM or $f(x) = \sum_{i=1}^{n} \tanh\left(x_{i-1} + x_i\right)$

Word2Vec Extension

- Le, Quoc V., and Tomas Mikolov. "**Distributed Representations of Sentences and Documents.**" *ICML* (2014).



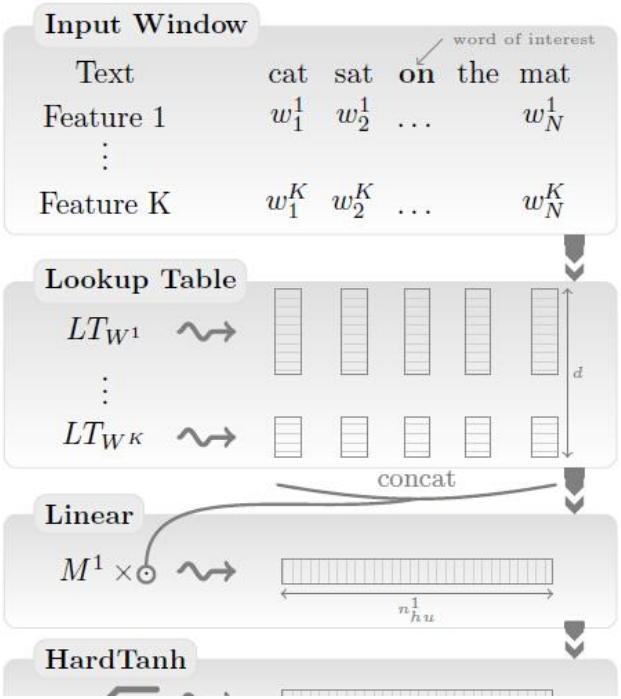


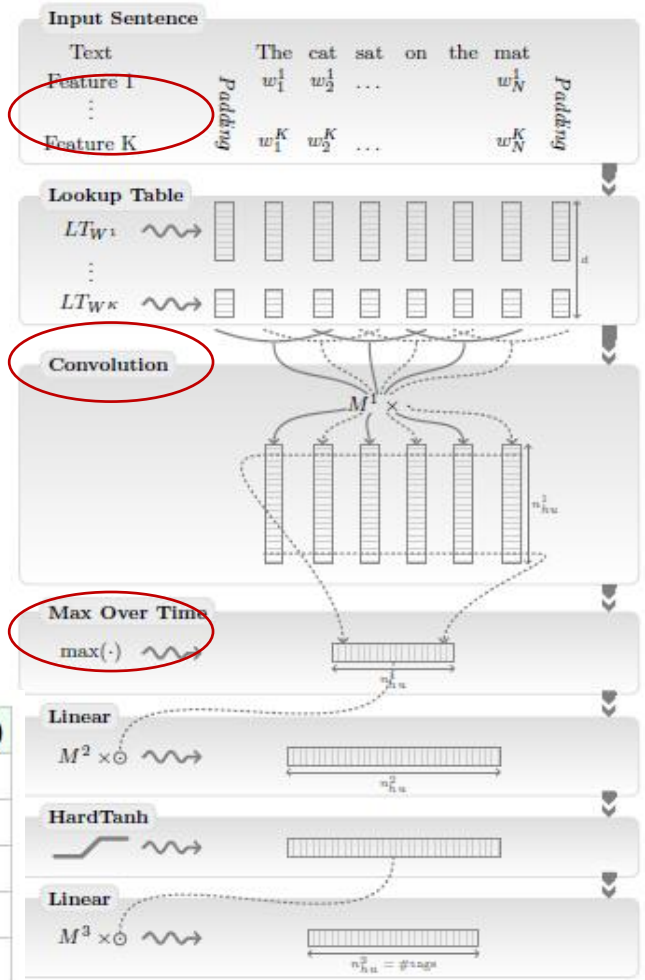
The paragraph token can be thought of as another word. It acts as a memory that remembers what is missing from the current context – or the topic of the paragraph

SENNA

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. "**Natural language processing (almost) from scratch**." *The Journal of Machine Learning Research* 12 (2011)

Word embeddings
ConvNet

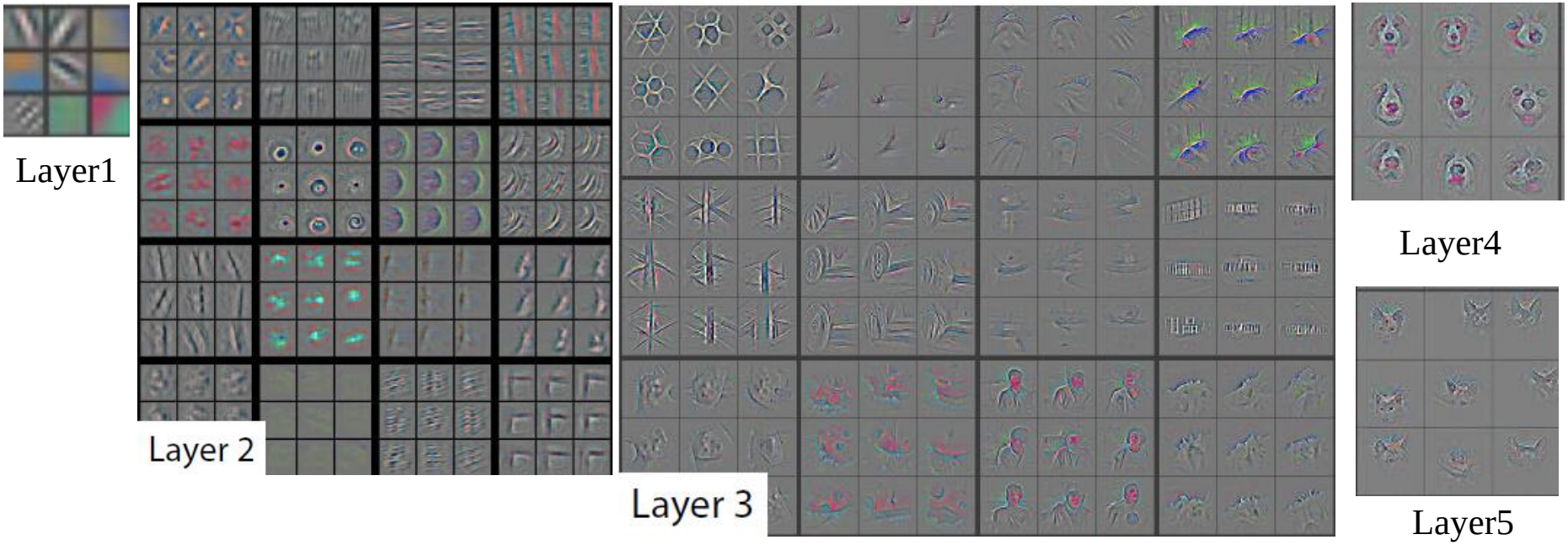




Task	Benchmark		Performance	Timing (s)
Part of Speech (POS)	(Toutanova et al, 2003)	(Accuracy)	97.29%	3
Chunking (CHK)	CoNLL 2000	(F1)	94.32%	2
Name Entity Recognition (NER)	CoNLL 2003	(F1)	89.59%	2
Semantic Role Labeling (SRL)	CoNLL 2005	(F1)	75.49%	36
Syntactic Parsing (PSG)	Penn Treebank	(F1)	87.92%	74

NN & Documents

- Le, Quoc V., and Tomas Mikolov. "**Distributed Representations of Sentences and Documents.**" *ICML* (2014).
- Denil, Misha, Alban Demiraj, Nal Kalchbrenner, Phil Blunsom, and Nando de Freitas. "**Modelling, Visualising and Summarising Documents with a Single Convolutional Neural Network**." *arXiv preprint arXiv:1406.3830* (2014).



How about Text?
- Word salience? – word/entity/topic extraction
- Sentence salience? – summarization

Zeiler, Matthew D., and Rob Fergus. "**Visualizing and understanding convolutional networks**." ECCV 2014

Multi-Document Summarization

- Liu, Yan, Sheng-hua Zhong, and Wenjie Li. "**Query-Oriented Multi-Document Summarization via Unsupervised Deep Learning**." In *AAAI*. 2012.

  - DBN
  - Query related weight

$$A_{ij}^1 = \max(A^1) \quad \text{if } v_i \in \mathbf{q}$$

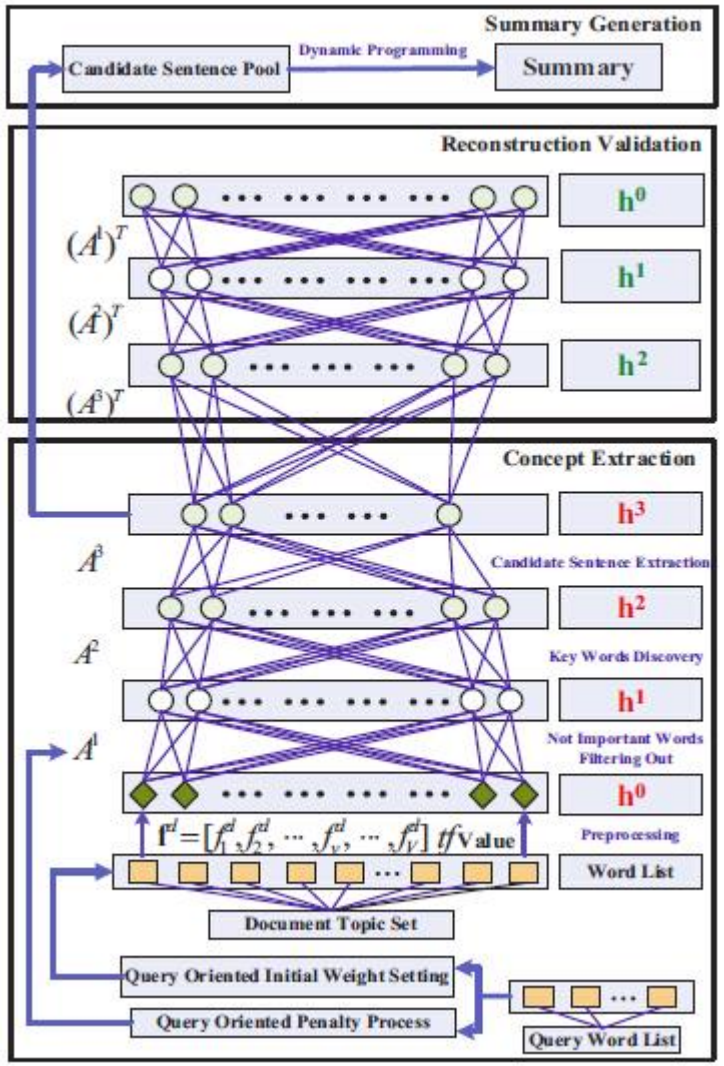
$$\Delta A_{ij}^1 = \gamma \Delta A_{ij}^1 \quad \text{if } v_i \in \mathbf{q}$$

  - Important words extraction

$$AF = [\underbrace{(\mathbf{f}^{D_1})^T, (\mathbf{f}^{D_2})^T, \cdots, (\mathbf{f}^{D_j})^T, \cdots, \mathbf{f}^{D})^T}_{K_3}](A^1 A^2 A^3)$$

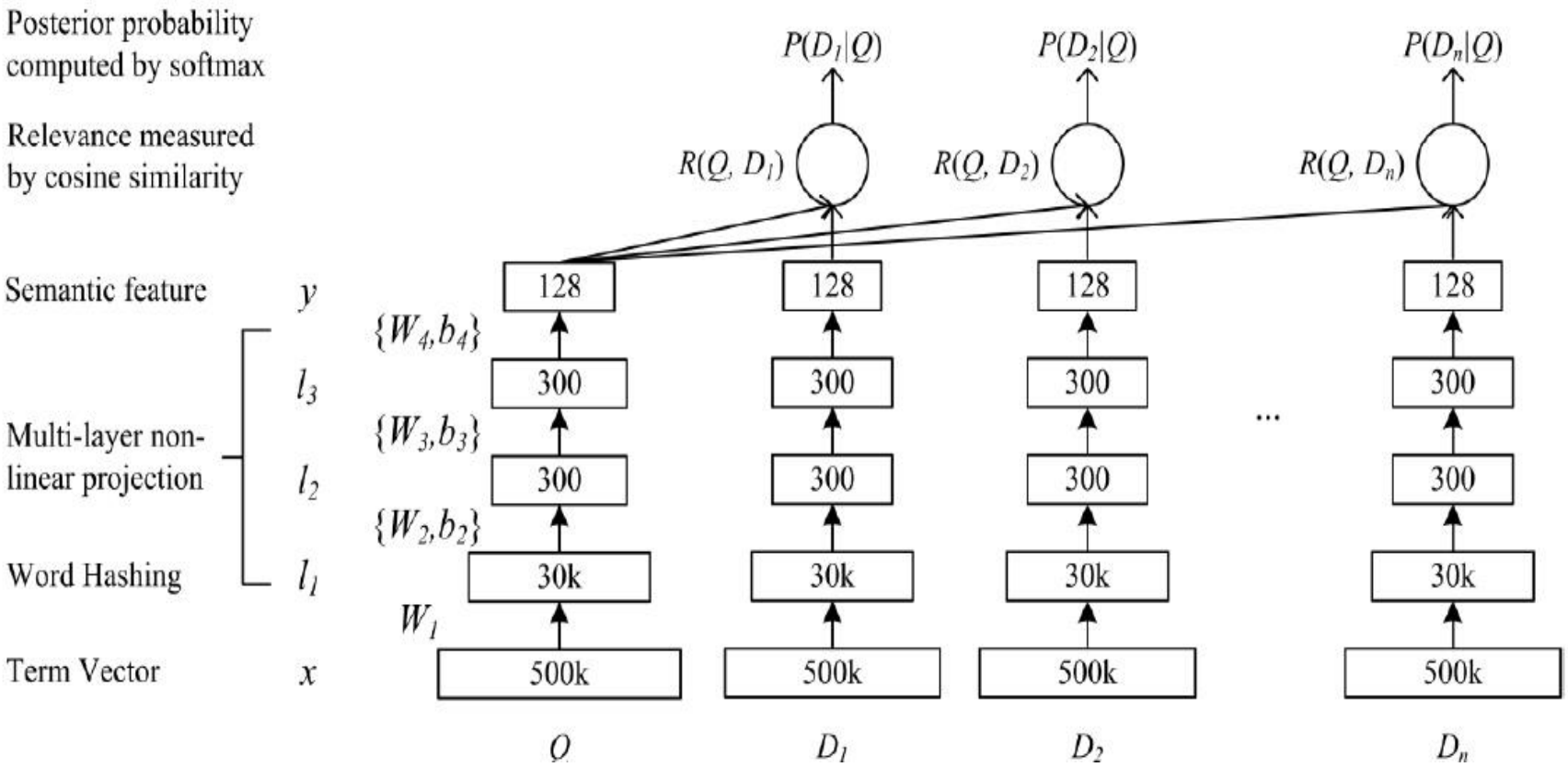
  - Sentence Selection

    Intersection with important words



Web Search | Learning to Rank

- Huang, Po-Sen, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. "**Learning deep structured semantic models for web search using clickthrough data**." CIKM, 2013.

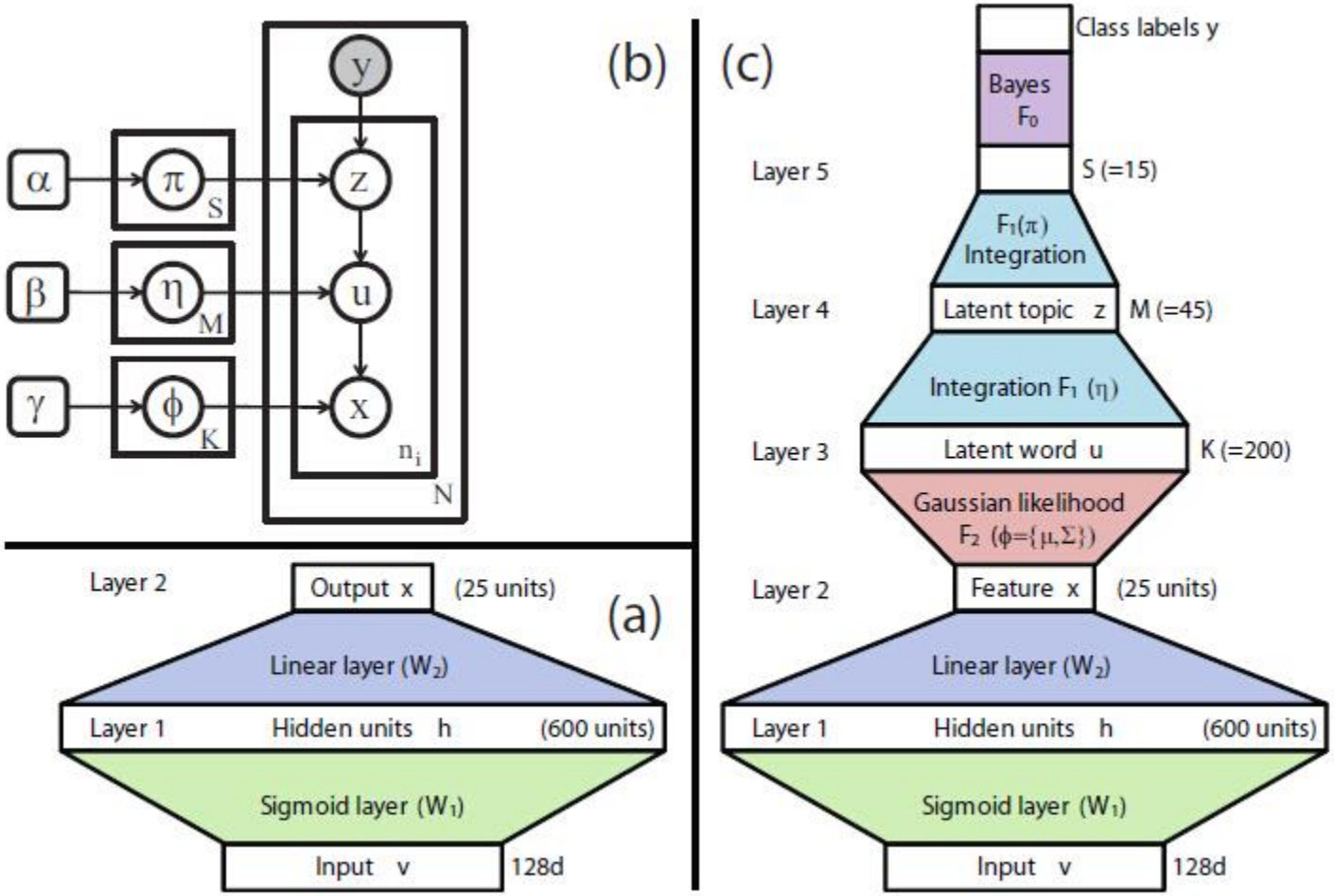


NN & Topic Model

- Wan, Li, Leo Zhu, and Rob Fergus. "**A hybrid neural network-latent topic model**." ICAIS. 2012.
- Larochelle, Hugo, and Stanislas Lauly. "**A neural autoregressive topic model**." *NIPS*. 2012.
- Hinton, Geoffrey E., and Ruslan Salakhutdinov. "**Replicated softmax: an undirected topic model**." NIPS 2009.
- Srivastava, Nitish, Ruslan R. Salakhutdinov, and Geoffrey E. Hinton. "**Modeling documents with deep boltzmann machines**." *UAI*(2013).
- Hinton, Geoffrey, and Ruslan Salakhutdinov. "**Discovering binary codes for documents by learning deep generative models**." *Topics in Cognitive Science* 3, no. 1 (2011): 74-91.
- Salakhutdinov, Ruslan, Joshua B. Tenenbaum, and Antonio Torralba. "**Learning to learn with compound hd models**." NIPS (2011).

Hybrid NN & Topic Model

- Wan, Li, Leo Zhu, and Rob Fergus. "**A hybrid neural network-latent topic model**." ICAIS. 2012.



Tuning

- #layers, #hidden units
- Learning rate
- Momentum
- Weight decay
- Mini-bach size
- Activation function (logistic, tanh, relu)

BE A GOOD TUNER

Thanks!

Q&A