Learning Representations

Parth Gupta

Amazon India

Http://www.dsic.upv.es/~pgupta

10 April, 2018 – ML class @ IIT Kharagpur

Representation learning

- So far, it is quite clear that deep learning is best suitable for learning abstract representations
- This part continuous to focus on how part!
- Two major categories:
  - Unsupervised
  - Supervised

Fill in the blanks

- indian institute of ___________
- times of ________
- hum hain raahi ________ ________

Fill in the blanks

- indian institute of technology
- times of india
- hum hain raahi pyar ke

Context is Important!!

N-gram Language models (LM)

- Assign probability to a sequence:
  - P(“indian statistical institute”)
    - P(“institute” | “indian statistical”)
    - count(“indian statistical institute”) / count(“indian statistical”)
  - P(“indian statistical institute”) > P(“indian statistical cinema”)
- 3-gram LM in terms of 2-gram LM
  - P(w1 w2 w3) = P(w3 | w1 w2) = P(w3 | w2) x P(w2 | w1)
- In general,

$$P\left(w_t | w_{t-n} w_{t-n+1} \cdots w_{t-1}\right)$$

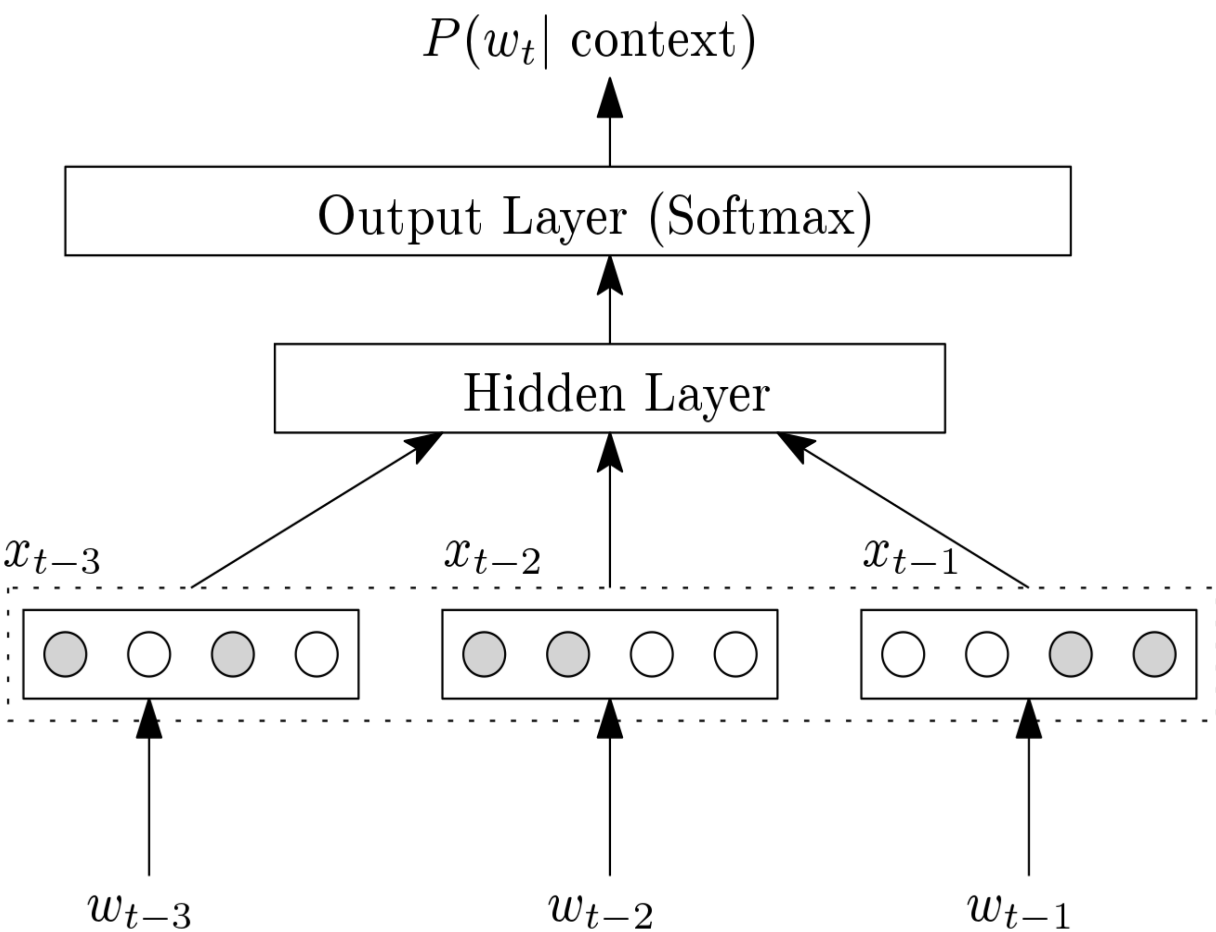
Generalisation

- Count generated from large corpus
- Would this generalise?
- If “cat is an animal” is there but “dog is an animal” is not. Can we still get P(“animal” | “dog is an”) to be the highest?
  - If this pattern is not completely present, may be partially present
    - “dog is”, “is an” “an animal” but it's difficult to generalise without knowing “dog” and “cat” has some semantic similarity!

Neural Network Language Model

Bengio et. al.
JMLR 2003

- Using NN, let's model $P(w_t | w_{t-n} w_{t-n+1} \cdots w_{t-1})$



Probability distribution
size of vocabulary

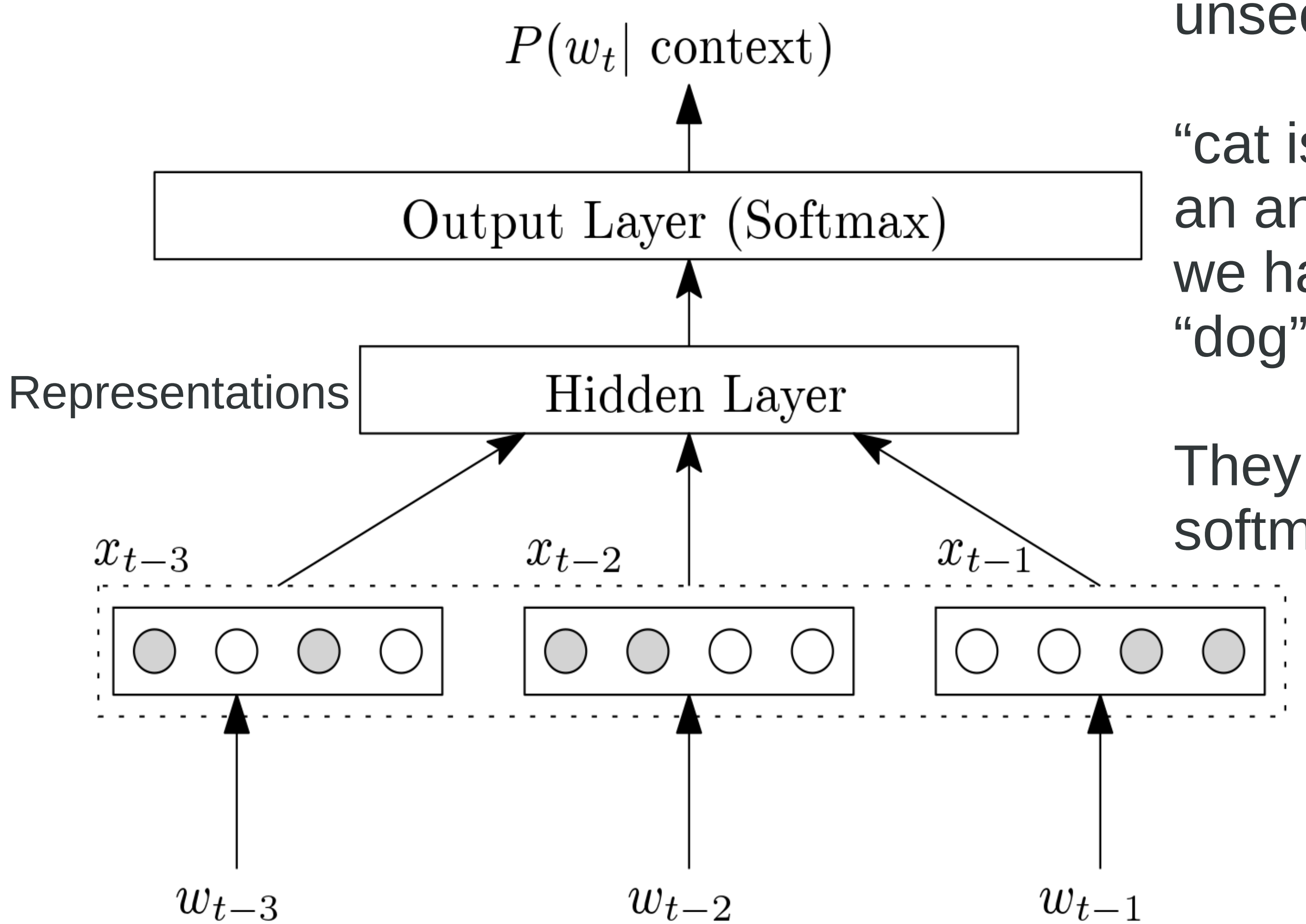
Addition

Continuous
representations

One-hot encoding
(fixed n=3)

Neural Network Language Model

Bengio et. al.
JMLR 2003



Potentially it will generalise to unseen patterns

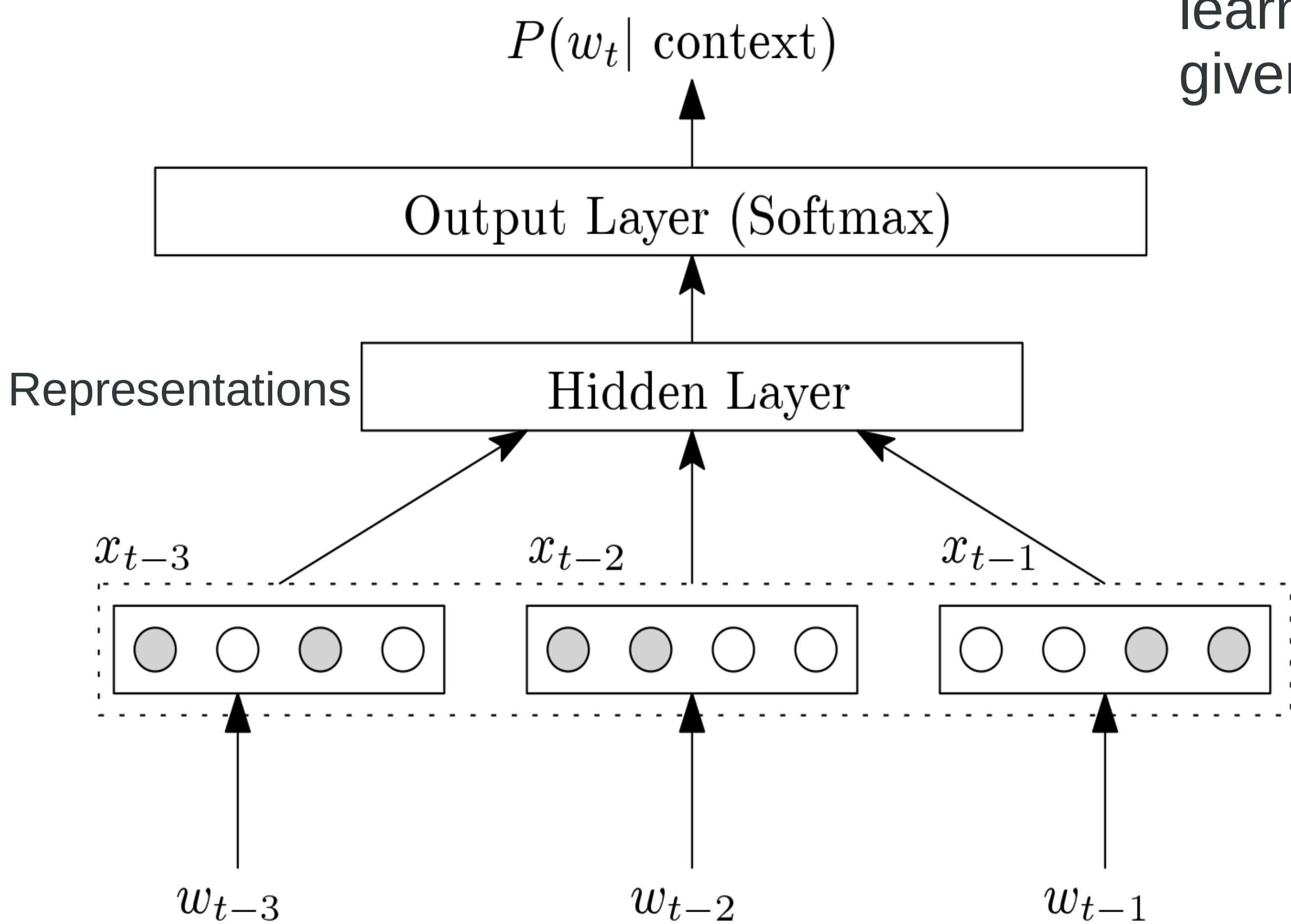
“cat is an animal” and “dog is an animal” is possible to get if we have vectors for “cat” and “dog” are somewhat similar.

They will saturate the correct softmax unit!

Neural Network Language Model

Bengio et. al.
JMLR 2003

In the training process, we learn representations for a given term in the hidden layer.



RNN Language Models

- Used to remove certain constraints for NNLM
- Variable length input
- Sometimes, RNNs provide more effective representation that NNs because of time dimension and Hidden-to-Hidden connection

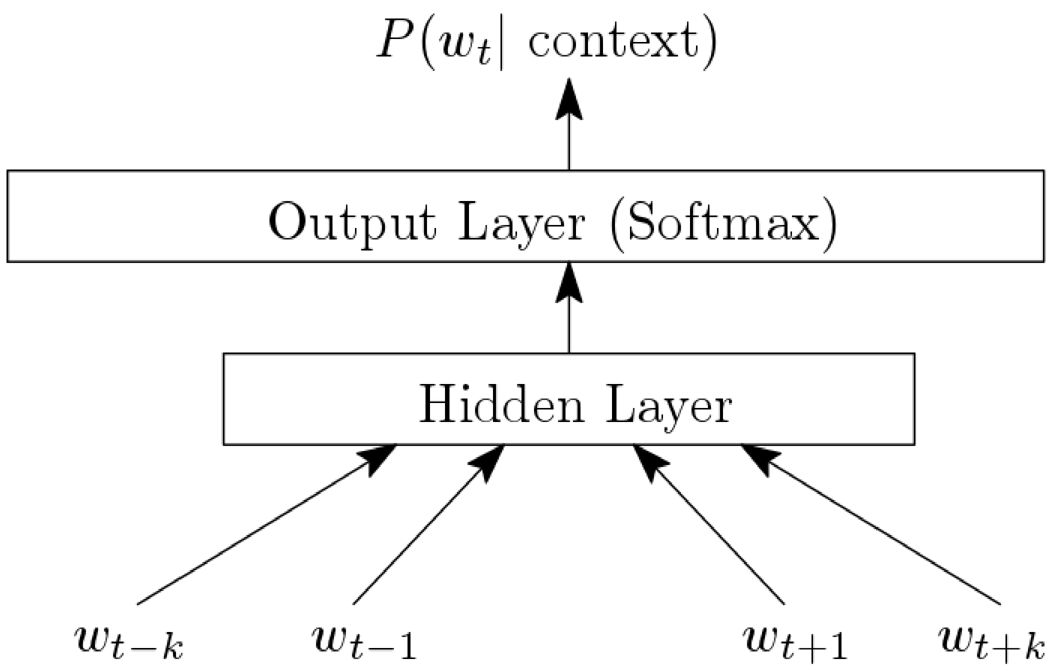
Word2Vec

- Certain improvements over NNLM and many tricks
- Effectively two types of models
  - Continuous Bag-of-Words (CBOW)
  - Skip-gram model (Skip-gram)

Continuous BOW

Mikolov et. al.
Arxiv 2013

- Direct one-hot input, no intermediate representations
- Trying to predict missing word from surrounding (variable length) context

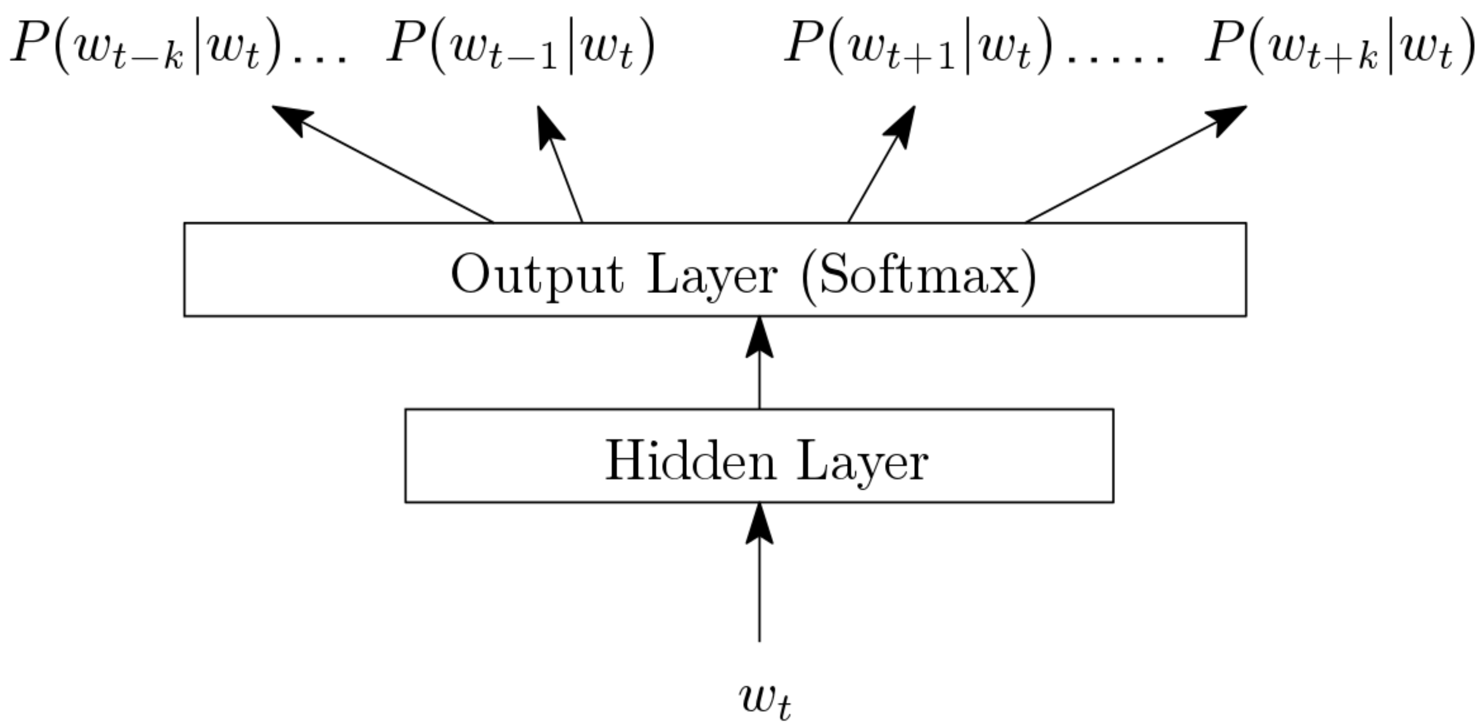


One-hot encoding

Mikolov et. al.
Arxiv 2013

Skip-gram Model

Predict the context given the word!



Softmax Output Layer

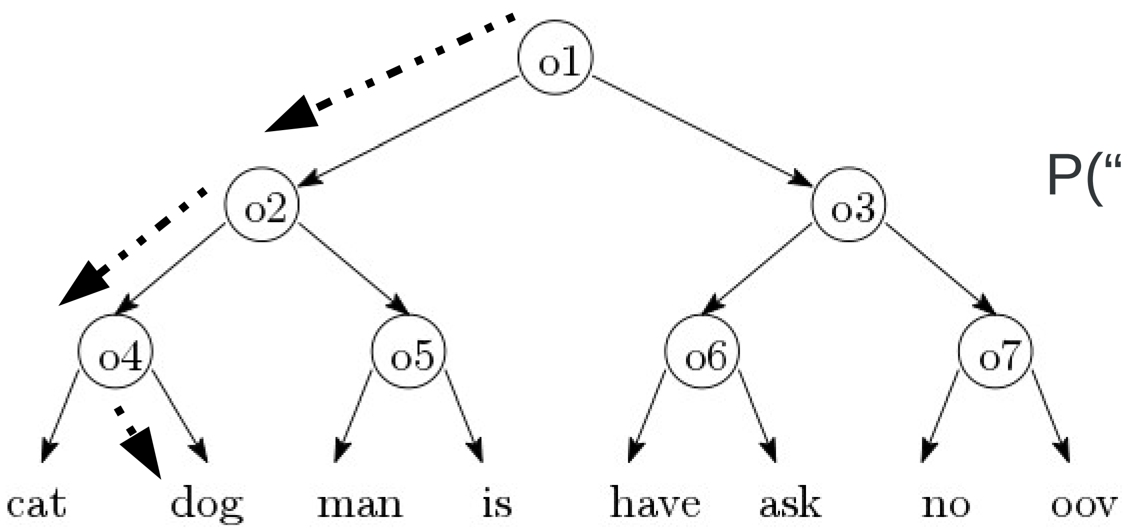
- Output layer probabilities $$p_j = \frac{e^{y_j}}{\sum_{i=1}^{V} e^{y_i}}$$
- Output layers from size 50k to 500k
- Quite heavy to compute
- Impractical for large vocabularies

Hierarchical Softmax

Bengio et. al.
JMLR 2003

- Rather than having a flat layer, consider it as a hierarchical layer where units represent the internal nodes of a binary tree
- Terms are at the leaf of a complete binary tree
- Unit value suggests to go towards left or right child
- Size of the layer = $\log_2(V)$

Significant improvement:
If V = 100000 → log(V) = 17



P(“dog” | context) = (1 – P(o1)
* 1 – P(o2)
* P(o4))

Creating Binary Tree

- Randomly
  - Random order
- Using Wordnet
  - Semantically similar words would be closer
  - Leads to significant improvements
- Hierarchical clustering
  - Tries to automatically cluster based on latent representations of the terms

Syntactic and Semantic relatedness

Mikolov et. al.
Arxiv 2013

Test collection of word pair similarities

	Type of relationship	Word Pair 1		Word Pair 2	
Semantic	Common capital city	Athens	Greece	Oslo	Norway
	All capital cities	Astana	Kazakhstan	Harare	Zimbabwe
	Currency	Angola	kwanza	Iran	rial
	City-in-state	Chicago	Illinois	Stockton	California
	Man-Woman	brother	sister	grandson	granddaughter
Syntactic	Adjective to adverb	apparent	apparently	rapid	rapidly
	Opposite	possibly	impossibly	ethical	unethical
	Comparative	great	greater	tough	tougher
	Superlative	easy	easiest	lucky	luckiest
	Present Participle	think	thinking	read	reading
	Nationality adjective	Switzerland	Swiss	Cambodia	Cambodian
	Past tense	walking	walked	swimming	swam
	Plural nouns	mouse	mice	dollar	dollars
	Plural verbs	work	works	speak	speaks

Mikolov et. al.
Arxiv 2013

Results

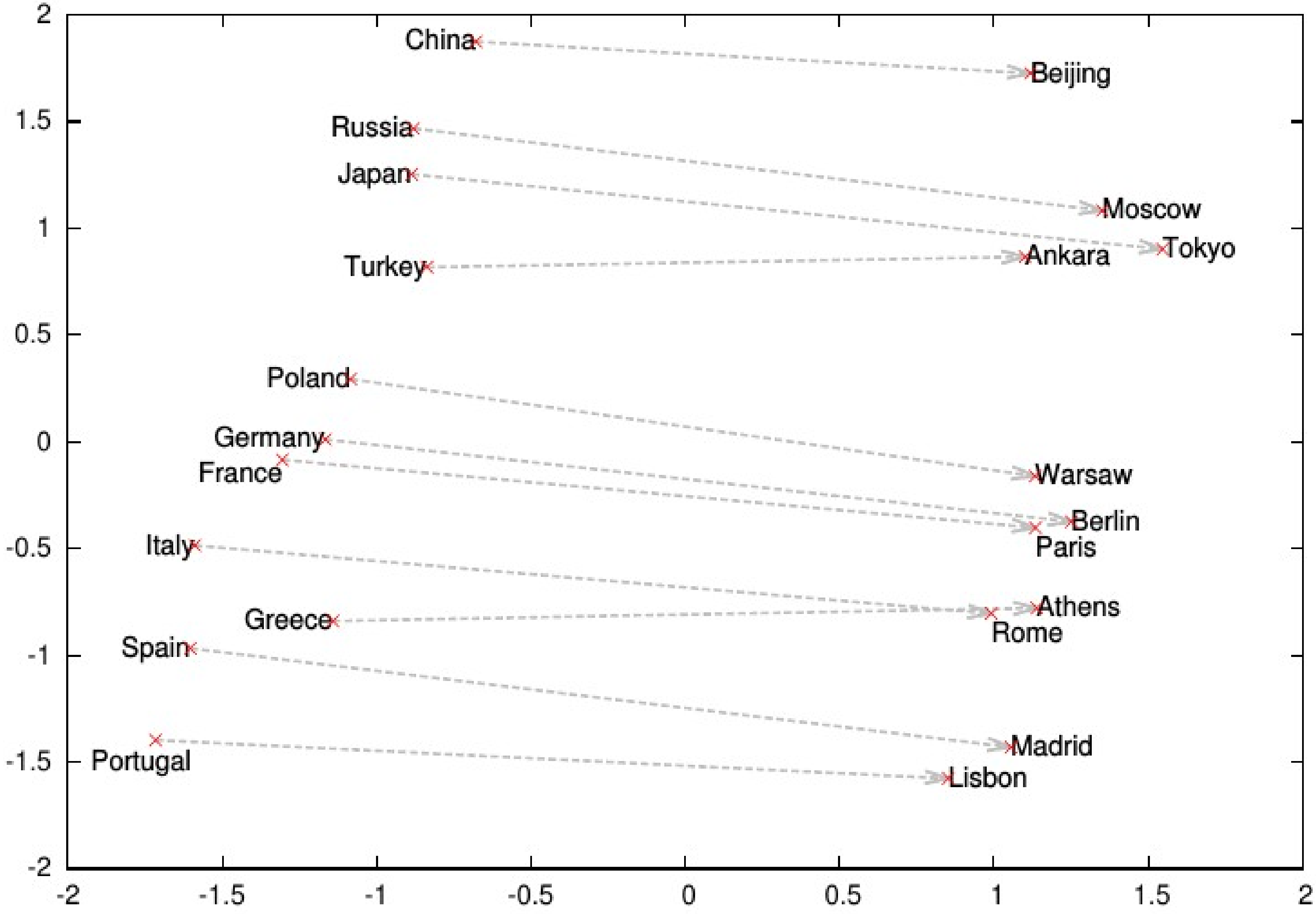
Model Architecture	Semantic-Syntactic Word Relationship test set		MSR Word Relatedness Test Set [20]
	Semantic Accuracy [%]	Syntactic Accuracy [%]	
RNNLM	9	36	35
NNLM	23	53	47
CBOW	24	64	61
Skip-gram	55	59	56

Word vector algebra

Paris – France + Italy = Rome

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

Projections



Mikolov et. al.
NIPS 2013

Learning Phrases

- Not often phrases are simple compositions of the constituting words
  - e.g. “new” + “york” + “times” != “new york times”
- Word to Phrases
  - Treat phrases as words, Simple!
- How to identify them?
  - Empirically from data
  - Pointwise mutual information

$$\frac{count(w_i, w_j) - delta}{count(w_i) * count(w_j)}$$

Mikolov et. al.
NIPS 2013

Examples

Find the fourth word given the three

Newspapers			
New York	New York Times	Baltimore	Baltimore Sun
San Jose	San Jose Mercury News	Cincinnati	Cincinnati Enquirer
NHL Teams			
Boston	Boston Bruins	Montreal	Montreal Canadiens
Phoenix	Phoenix Coyotes	Nashville	Nashville Predators
NBA Teams			
Detroit	Detroit Pistons	Toronto	Toronto Raptors
Oakland	Golden State Warriors	Memphis	Memphis Grizzlies
Airlines			
Austria	Austrian Airlines	Spain	Spainair
Belgium	Brussels Airlines	Greece	Aegean Airlines
Company executives			
Steve Ballmer	Microsoft	Larry Page	Google
Samuel J. Palmisano	IBM	Werner Vogels	Amazon

Mikolov et. al.
NIPS 2013

Frequent words

- How often there would be terms like “the”, “a”, “and” appear in the training in a (very) large corpora?
- More meaningful context for “India”? → “Delhi” vs. “the”
- Also vectors of such frequent terms don't change much during the training
- Hence, sub-sample them
  - Discard a training example associated with a word with probability p = function(TF(w_i)) where TF(w_i) is freq. of w_i

Mikolov et. al.
NIPS 2013

Compositions

- In word2vec, word-vectors are added to form the context to maximize average log probability:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t)$$

Hence if some words (e.g. “PM” “India”) appears quite often in the context for the given word “Narendra Modi”, this would lead to additive compositions like

PM + India = Narendra Modi

Mikolov et. al.
NIPS 2013

Composition Results

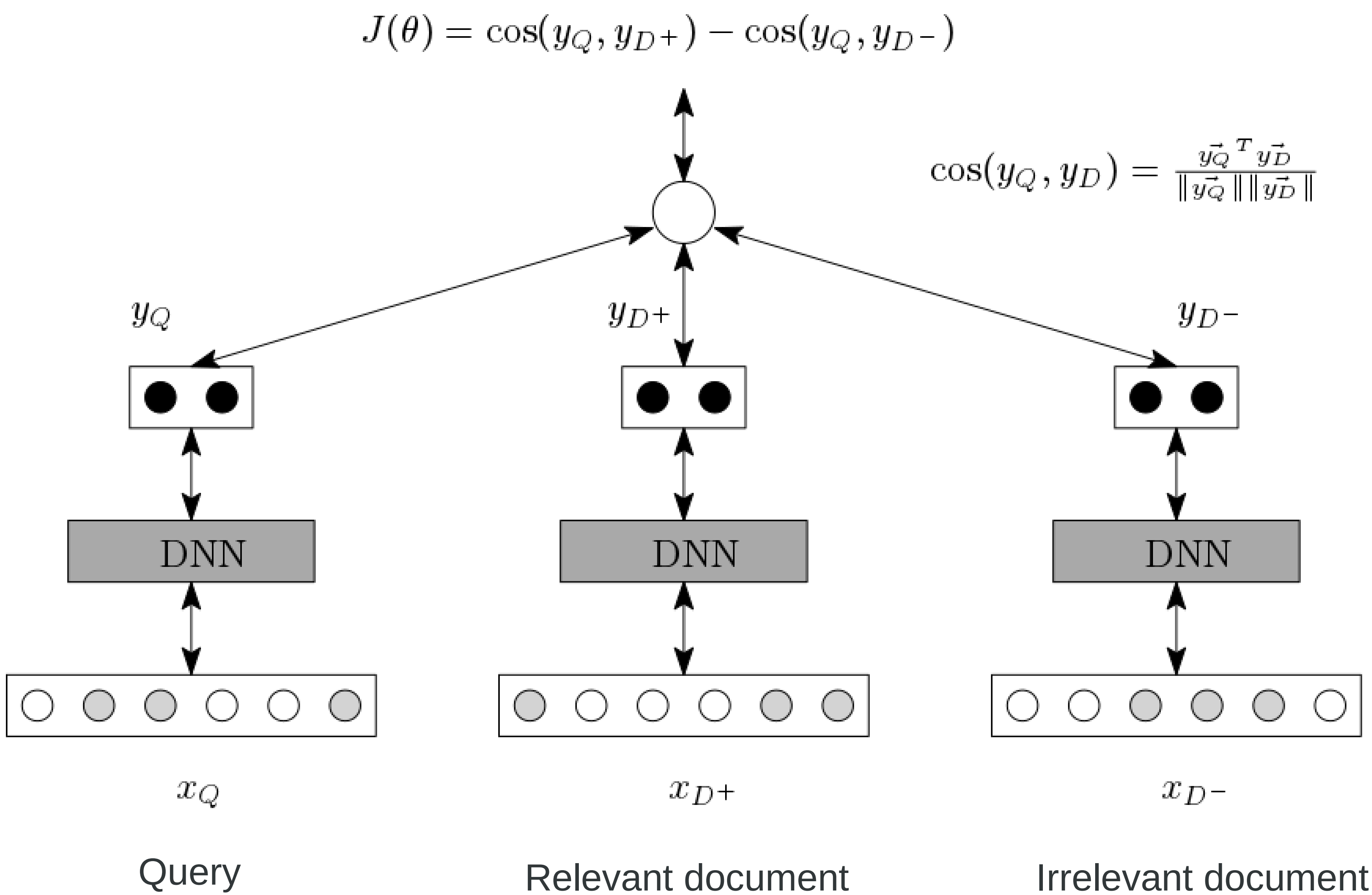
Closest tokens for the given addition

Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

DSSM: distributed structured semantic model

- So far, the training has been unsupervised – i.e. we don't tell the model explicitly that these two words have closer meanings or these two text are semantically similar
- Sometimes, we do have such information
- User clicks in web-search
  - Query-document pairs
  - We have some relevance signals

DSSM


$J(\theta) = \cos(y_Q, y_{D+}) - \cos(y_Q, y_{D-})$
$\cos(y_Q, y_D) = \frac{\vec{y_Q}^T \vec{y_D}}{\|\vec{y_Q}\| \|\vec{y_D}\|}$
y_Q
y_{D+}
y_{D-}
DNN
DNN
DNN
x_Q
x_{D+}
x_{D-}
Query
Relevant document
Irrelevant document

Training DSSM

- Calculate the gradient of J(θ) and backpropagate in the network
- Error function forces such representations which maximises the cosine similarity between the query and relevant document
- Noise contrastive component: It also tries to minimise the cosine similarity between the query and a irrelevant document

Huang et. al.
CIKM 2013

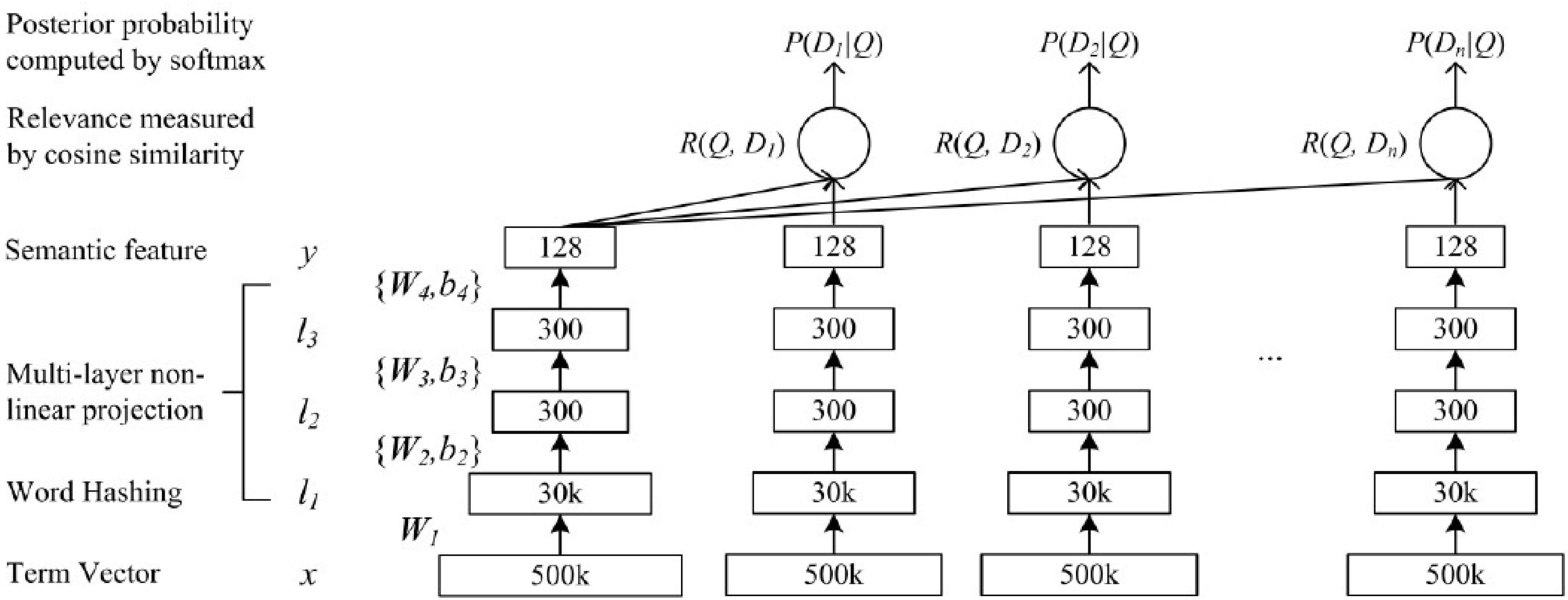
Word hashing

- For web search the vocabulary can really go high!
- Many valid non-language terms e.g. “www”, “y2k”, “iphone”, “i7”
- Encode the vocabulary into bag-of-character-grams
- “y2k” will become a combination of word-hashes “#y2”, “y2k”, “2k#” where '#' is marking the term boundary
- So now the vocabulary is all the word-hashes

Drastic compression
500k words → 30k word hashes

DSSM

Huang et. al.
CIKM 2013



Huang et. al.
CIKM 2013

Results

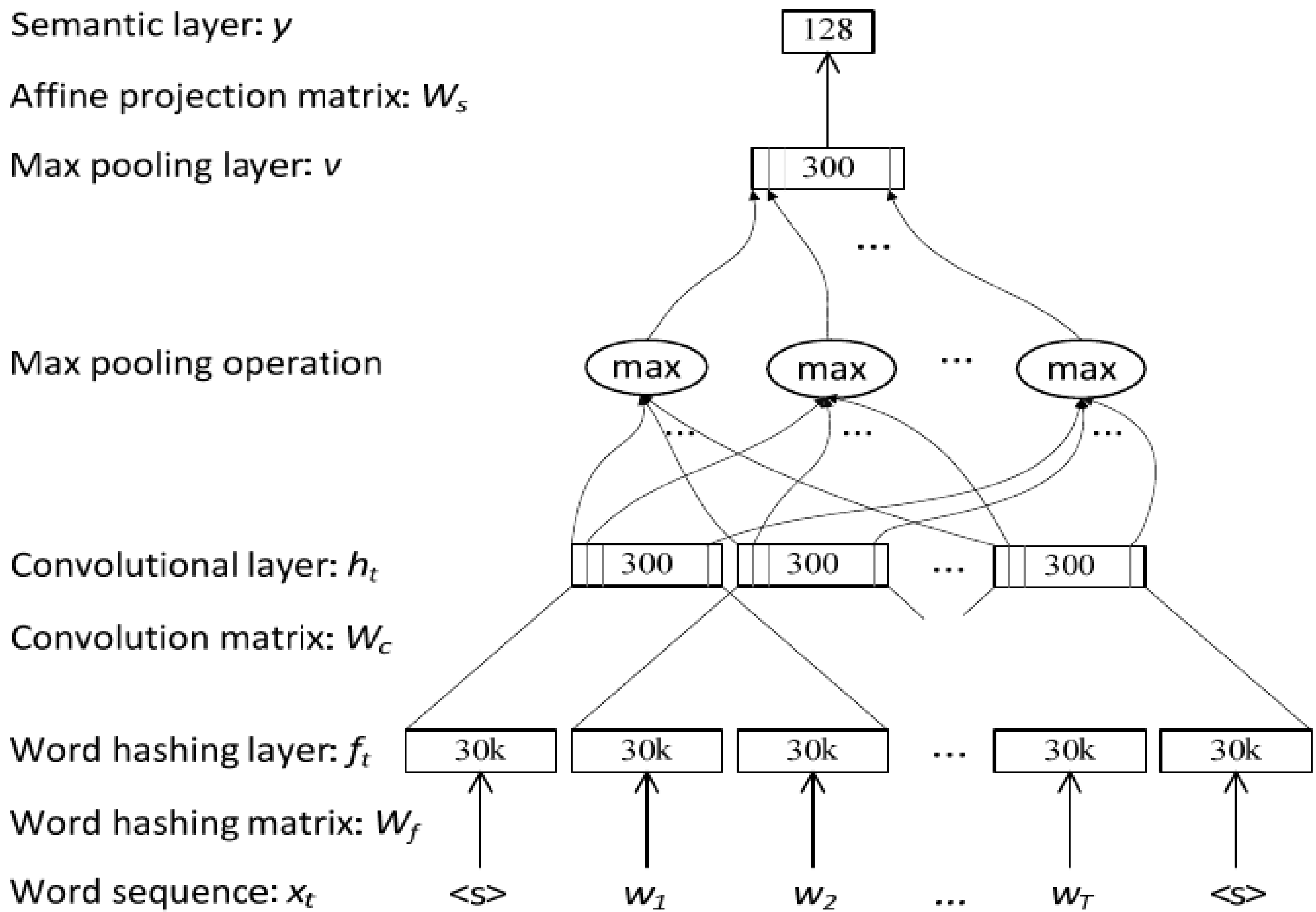
Test collection of 16k query and document title pairs

Vocabulary = 40k except WH

#	Models	NDCG@1	NDCG@3	NDCG@10
1	TF-IDF	0.319	0.382	0.462
2	BM25	0.308	0.373	0.455
3	WTM	0.332	0.400	0.478
4	LSA	0.298	0.372	0.455
5	PLSA	0.295	0.371	0.456
6	DAE	0.310	0.377	0.459
7	BLTM-PR	0.337	0.403	0.480
8	DPM	0.329	0.401	0.479
9	DNN	0.342	0.410	0.486
10	L-WH linear	0.357	0.422	0.495
11	L-WH non-linear	0.357	0.421	0.494
12	**L-WH DNN**	**0.362**	**0.425**	**0.498**

CDSSM

Shen et. al.
WWW 2014



CDSSM Results

Shen et. al.
WWW 2014

#	Models	NDCG@1	NDCG@3	NDCG@10
1	BM25	0.305	0.328	0.388
2	ULM	0.304	0.327	0.385
3	WTM	0.315^{α}	0.342^{α}	0.411^{α}
4	PTM (len $\leq$ 3)	0.319^{α}	0.347^{α}	0.413^{α}
5	DSSM	0.320^{α}	$0.355^{\alpha\beta}$	$0.431^{\alpha\beta}$
6	**C-DSSM win =3**	$\mathbf{0.342}^{\alpha\beta\gamma}$	$\mathbf{0.374}^{\alpha\beta\gamma}$	$\mathbf{0.447}^{\alpha\beta\gamma}$

Questions?

Thanks!

Story so far..

- Basics of Deep Learning
- Deep Learning Architectures and Frameworks
- Learning Representations
  - Neural Network Language Model
  - Word2Vec (Continuous BoW, Skip-gram)
  - Learning Phrases
  - DSSM
  - CDSSM
- Applications of Deep Learning for IR
- Summary

Thanks!