Microsoft Research

Faculty Summit

2014 15TH ANNUAL

Microsoft Research Faculty Summit 2014 15TH ANNUAL

Deep Semantic Similarity Model for Text Processing

Presented by Xiaodong He and Jianfeng Gao

Microsoft

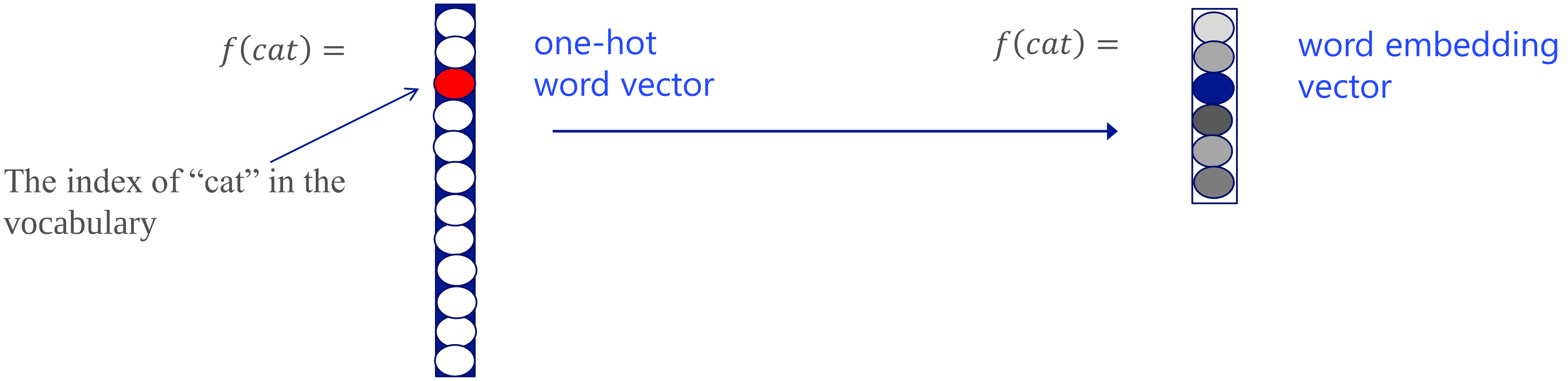
DSSM for learning the semantic meaning of texts

Learning the semantic meaning of texts is a key problem in NLP

Semantic Embedding

Word embedding: representing the meaning of a word by a vector

From discrete symbolic representation to continuously-valued vector representation

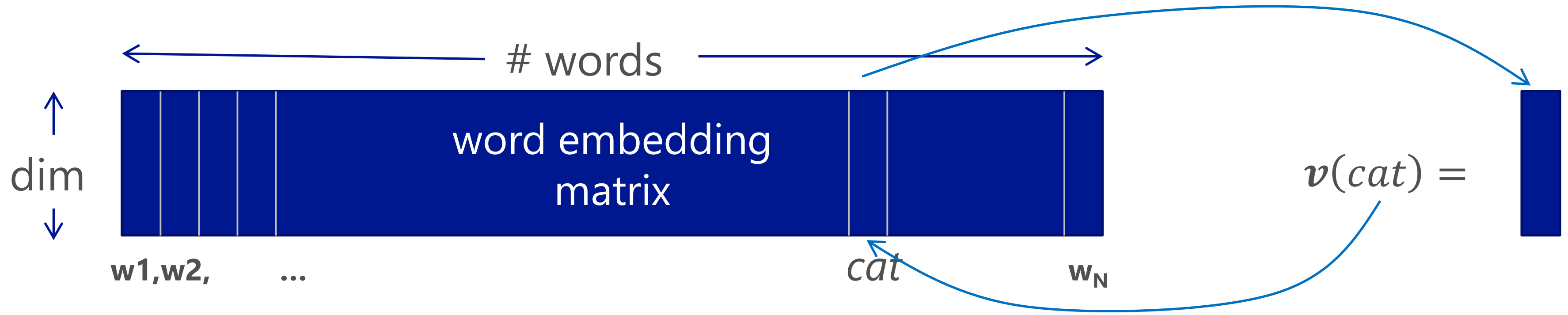


Common neural network based word embedding approaches

(Bengio 2001; Schwenk et al., 2006; Collobert et al., 2011; Mikolov et al. 2011, 2013, etc.)

Beyond Word Embedding

Word embedding: *one vector per word*



However, a decomposable, robust representation is preferable for large scale NL tasks

- New words, misspellings, and word fragments frequently occur (*generalizability*)
- Vocabulary of real-world big data tasks could be huge (*scalability*)
  - e.g., 100M+ unique words in a modern commercial search engine log

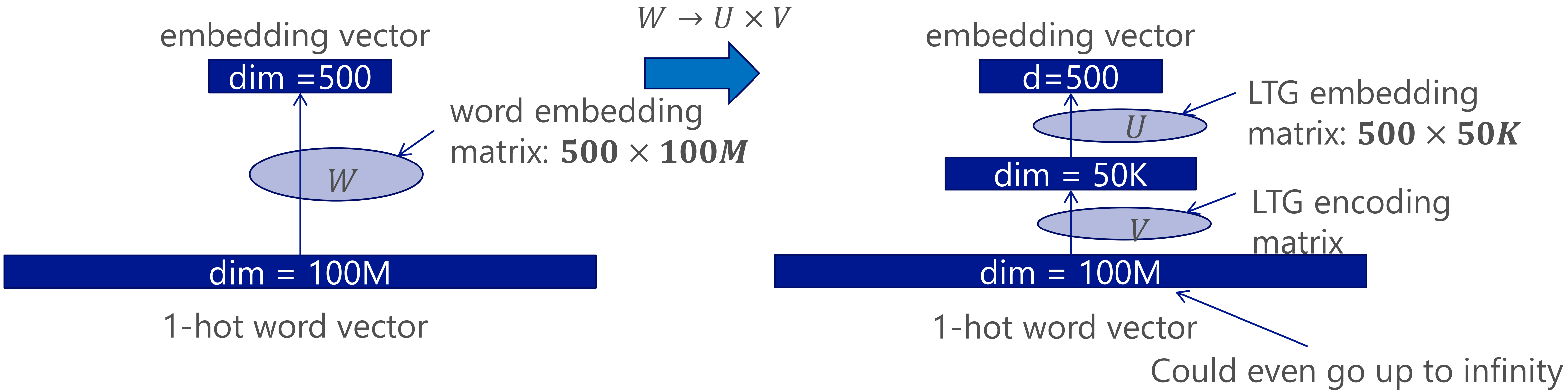
From Word to Sub-word Unit

Decompose word to sub-word units, e.g., letter-trigram (LTG)

cat → #cat# → #-c-a, c-a-t, a-t-#

Unbounded variability (word) => bounded variability (sub-word)

E.g., only ~50K letter-trigrams in English (37^3)

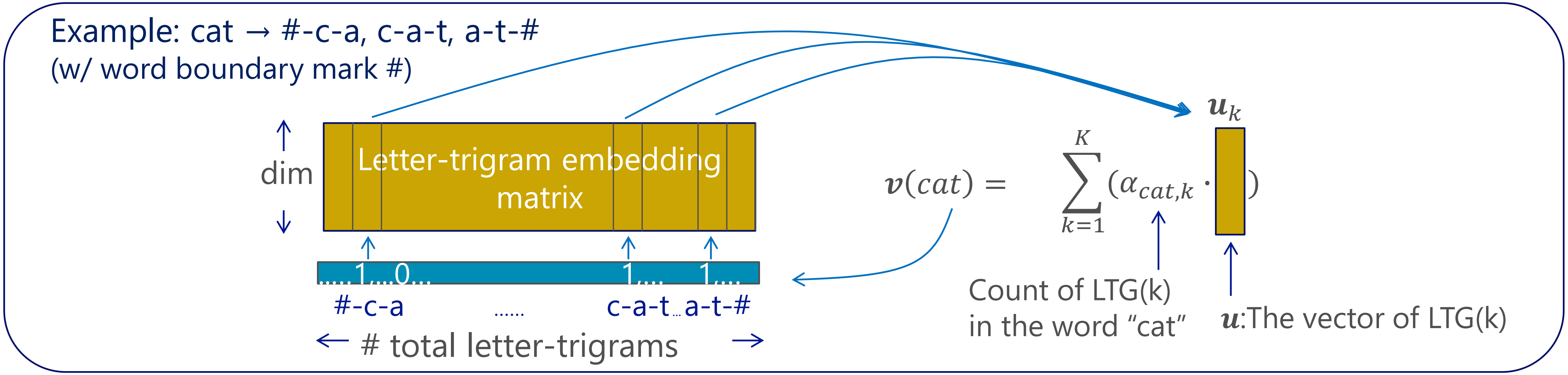


[Huang, He, Gao, Deng, Acero, Heck, CIKM2013]

Letter-trigram as the Sub-word Unit

Learn *one vector per letter-trigram* (LTG), the encoding matrix is a fixed matrix

Use the count of each LTG in the word for encoding



- Address both the *scalability* and *generalizability* issues

Semantic Embedding: from Word to Phrase

- The semantic intent is better defined at the phrase/sentence level rather than at the word level
  - The meaning of a single word is often ambiguous
  - A phrase/sentence/document contains rich contextual information that could be leveraged

DSSM for Semantic Embedding Learning

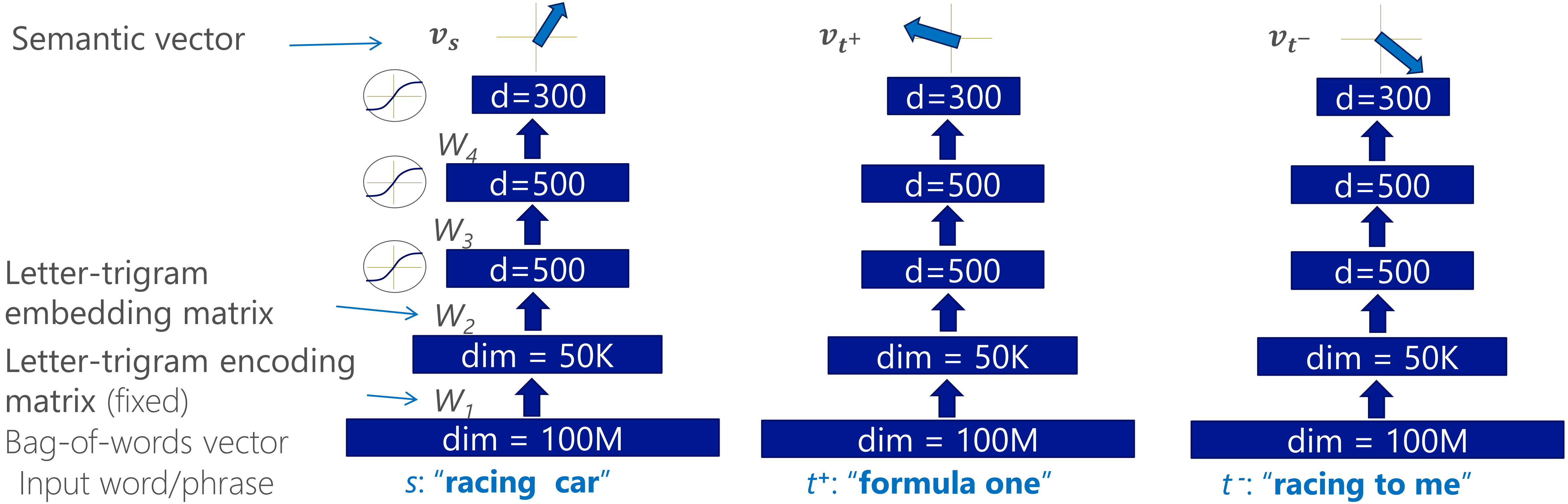
- Deep structured semantic model/Deep semantic similarity model (DSSM)
  - The DSSM refers to a series of **deep** semantic models developed recently at MSR
    - With variations on model structures and training objectives
  - The DSSM is trained by an **semantic similarity-driven objective**
    - projecting semantically similar phrases to vectors close to each other
    - projecting semantically different phrases to vectors far apart
  - The DSSM uses the **letter-trigram** sub-word vector for the input word representation

[Huang, He, Gao, Deng, Acero, Heck, CIKM2013]
[Shen, He, Gao, Deng, Mesnil, WWW2014]
[Gao, He, Yih, Deng, ACL2014]
[Yih, He, Meek, ACL2014]
[He, Gao, Deng, ICASSP2014 Tutorial]

DSSM for Semantic Embedding Learning

Initialization:

Neural networks are initialized with random weights

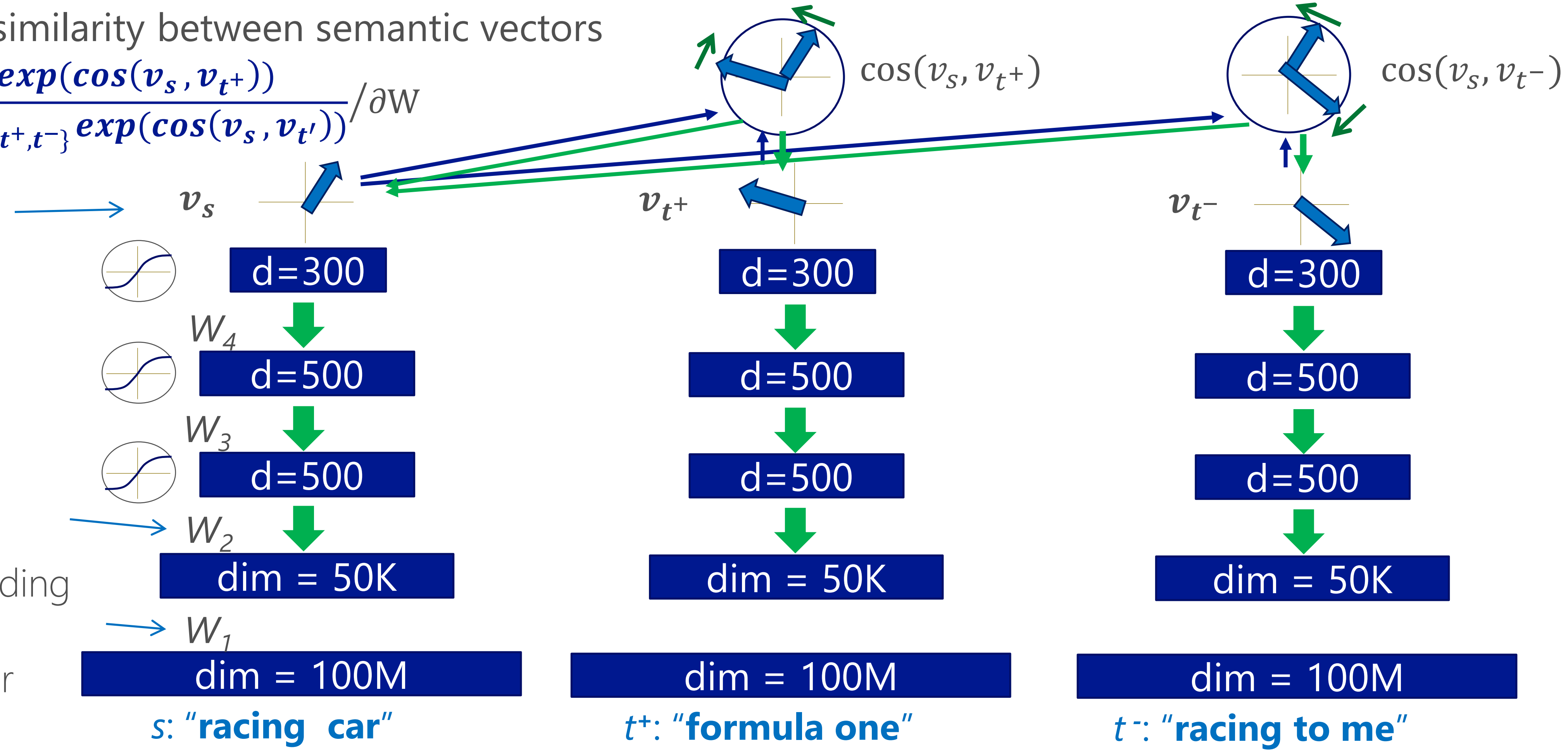


DSSM for Semantic Embedding Learning

Training:

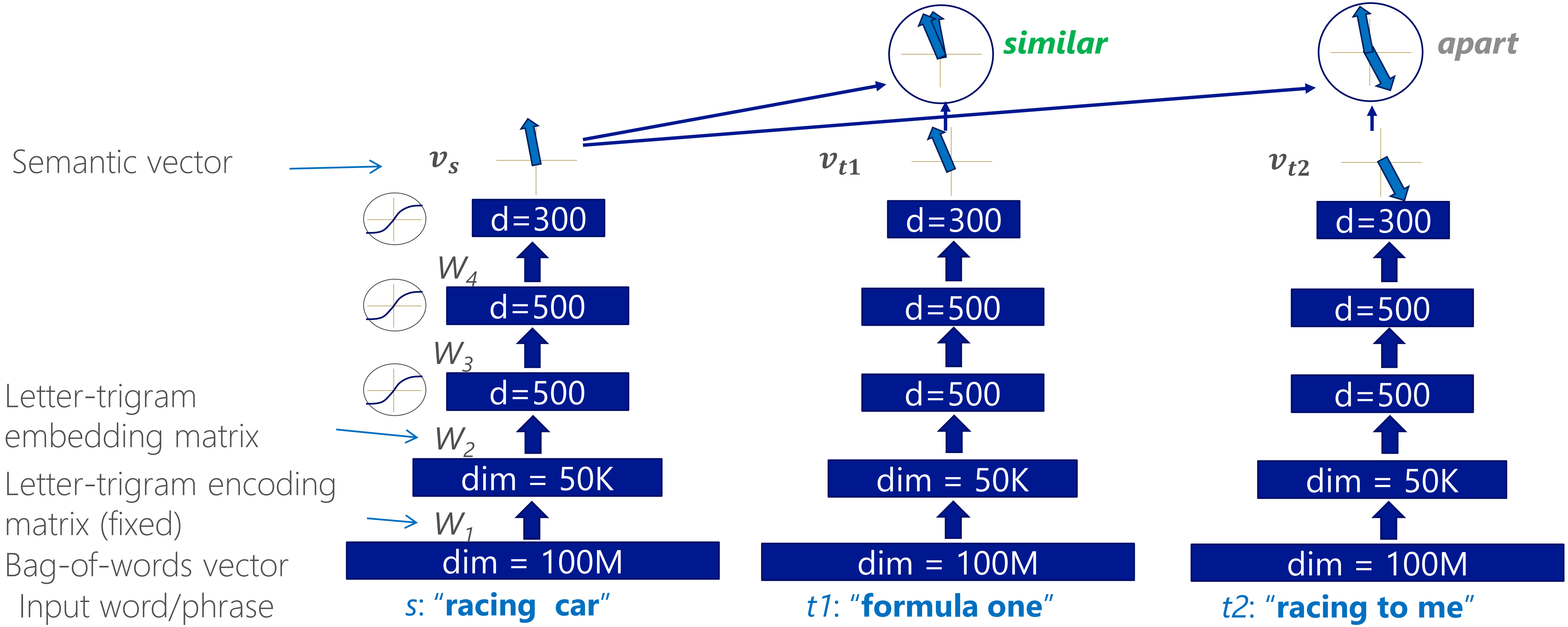
Compute Cosine similarity between semantic vectors

Compute gradients $\partial \dfrac{\boldsymbol{exp(cos(v_s, v_{t^+}))}}{\sum_{t'=\{t^+,t^-\}} \boldsymbol{exp(cos(v_s, v_{t'}))}} / \partial W$



DSSM for Semantic Embedding Learning

Runtime:



Evaluation

Evaluated on a information retrieval task

Docs are ranked by the cosine similarity between semantic vectors of the query and the doc

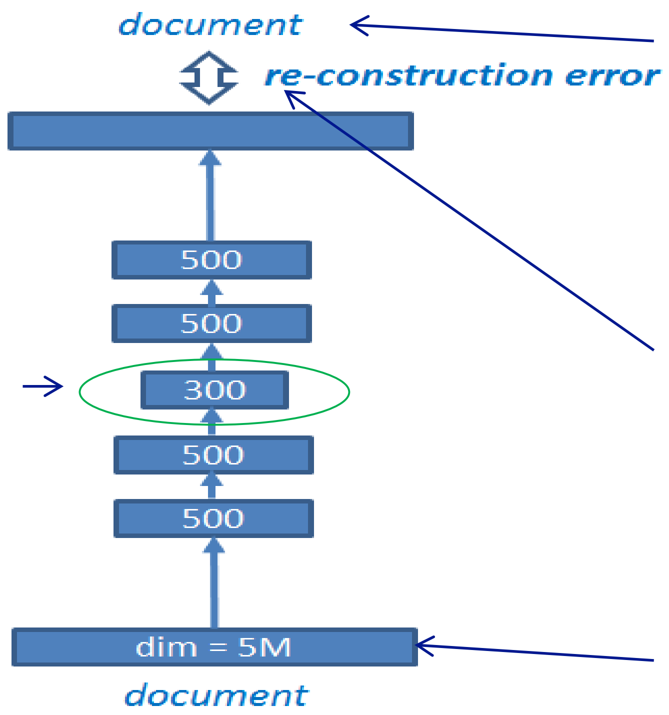
Model	Input dimension	NDCG@1 %
BM25 baseline	--	30.8
Probabilistic LSA (PLSA)		29.5
Auto-Encoder (Word)	40K	31.0 (+0.2)
DSSM (Word)	40K	34.2 (+3.4)
DSSM (Random projection)	30K	35.1 (+4.3)
DSSM (Letter-trigram)	30K	36.2 (+5.4)

DSSM-based embedding improves 5~7 pt NDCG over shallow models

The higher the NDCG score the better, 1% NDCG difference is statistically significant.

Comparison: Auto-encoder vs. DSSM

Auto-encoder



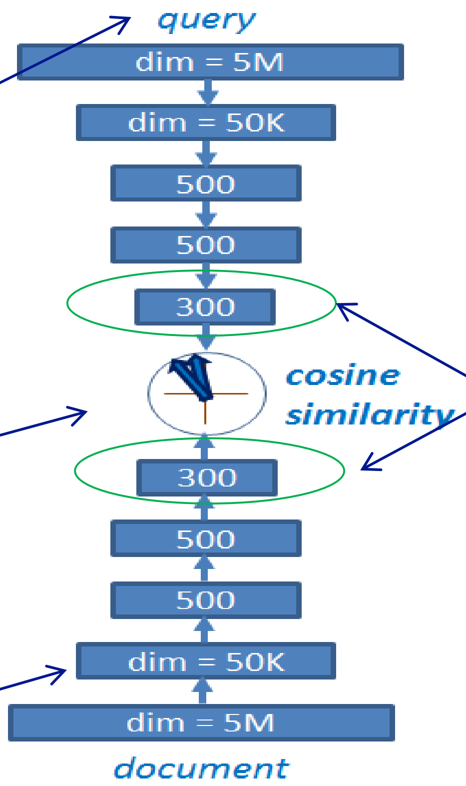
Embedding

Supervision:
AE: unsupervised
(e.g., doc<->doc)
DSSM: weakly supervised
(e.g., query<->doc search log)

Training objective:
AE: reconstruction error
DSSM: distance between embedding vectors

Input representation:
AE: 1-hot word vector
DSSM: letter-trigram

DSSM



Embedding

The DSSM can be trained using a variety of signals without costly labeling effort (e.g., user behavior log data).

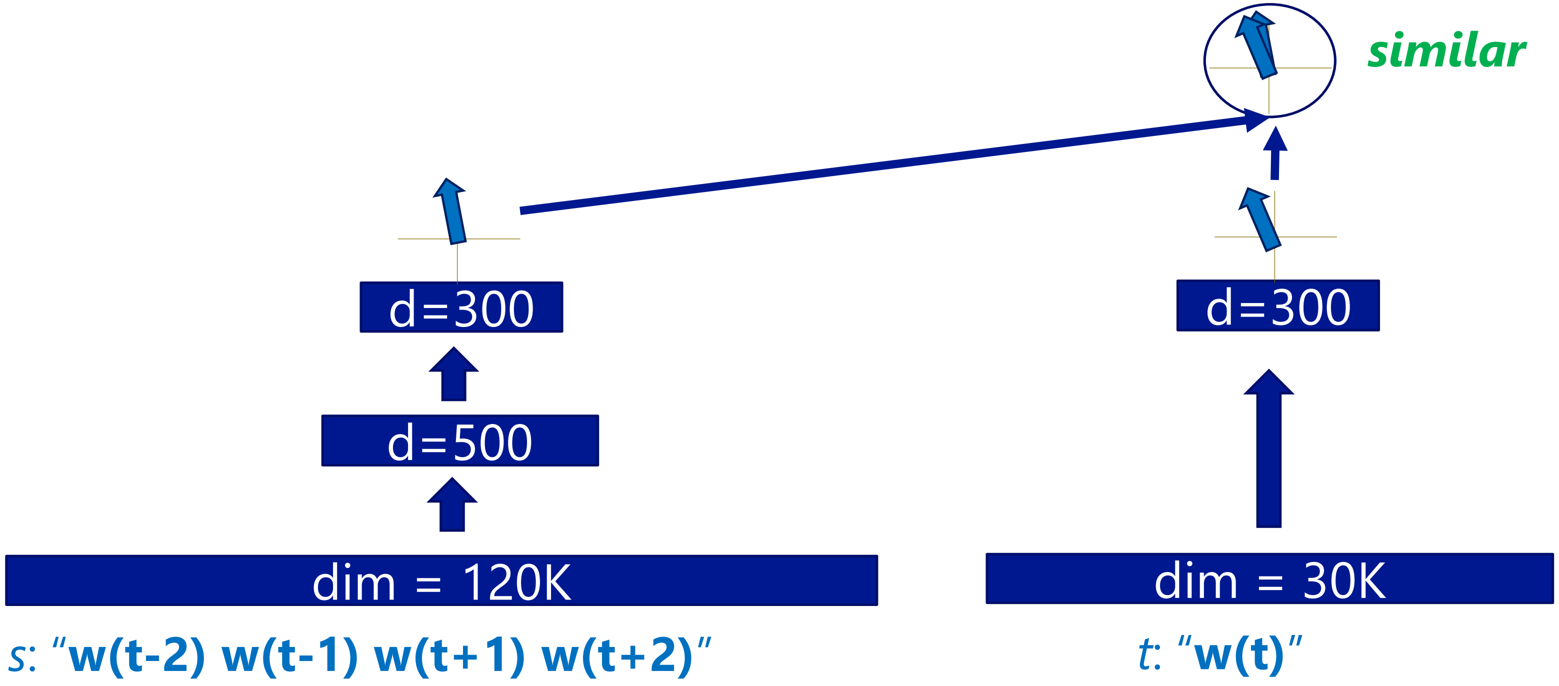
DSSM for Semantic Word Clustering and Analogy

Learn word embedding by means of its neighbors (context)

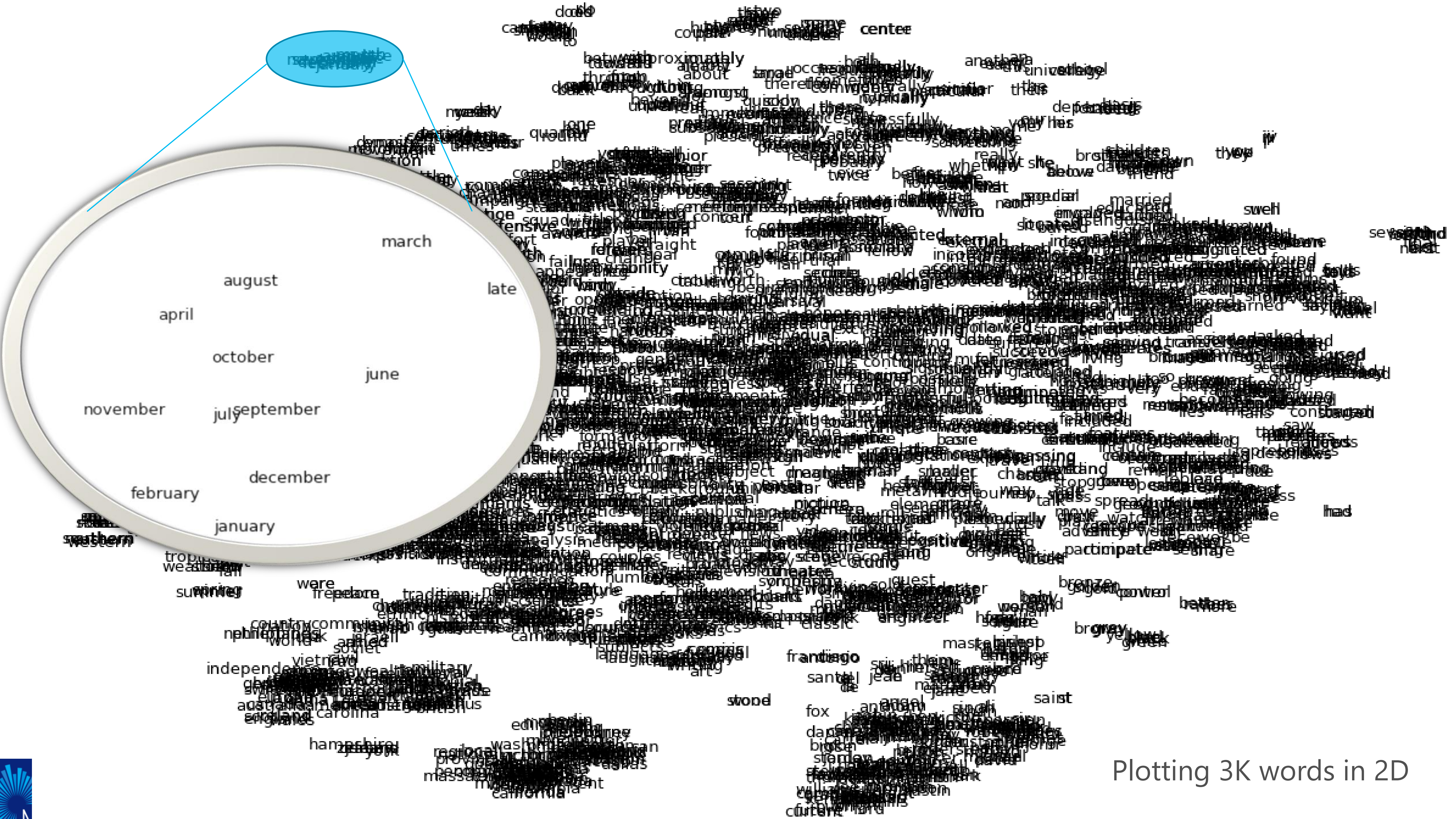
Construct context <-> word training pair for DSSM

Training Condition:

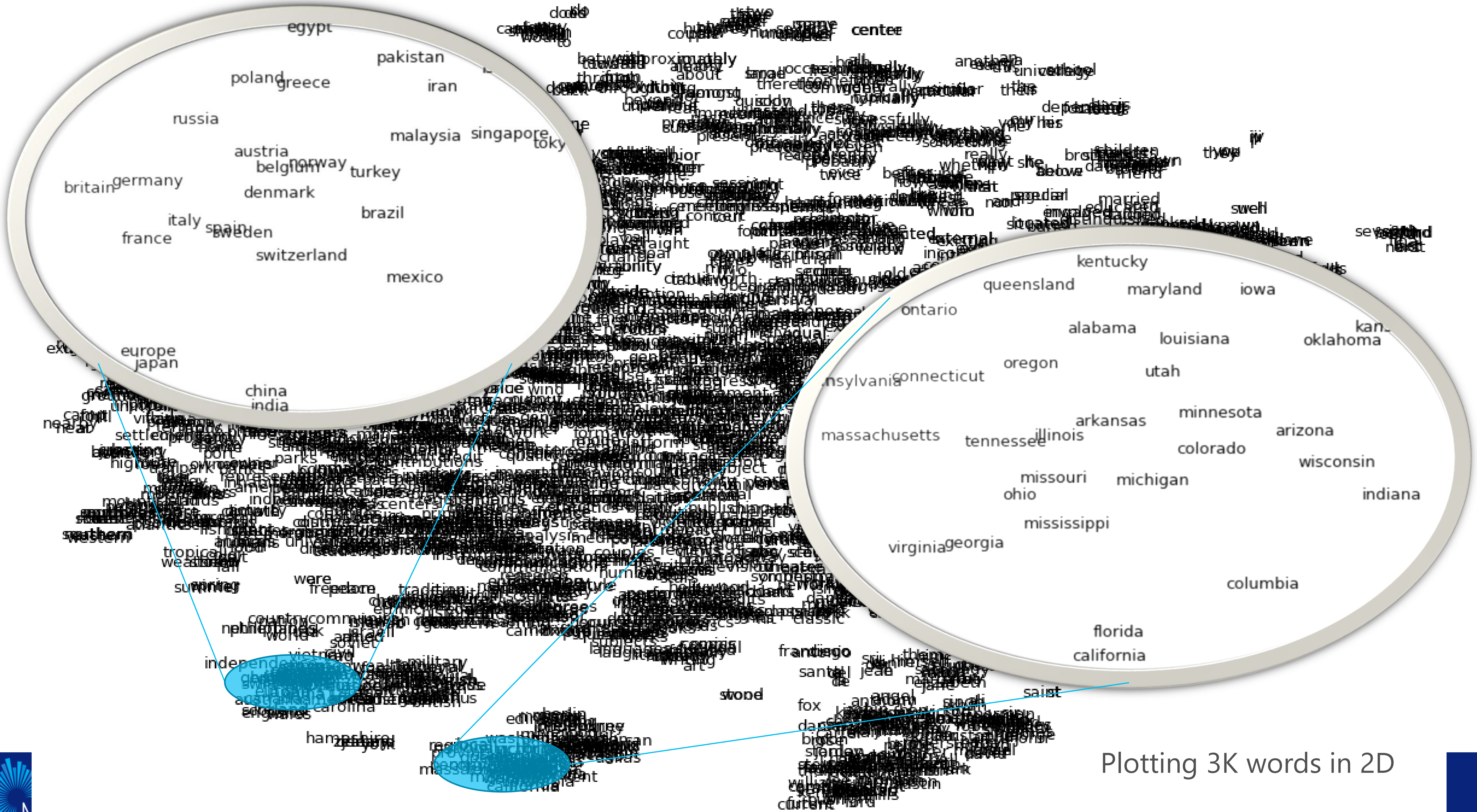
30K vocabulary size
10M words from Wikipedia
50-dimentional vector
Pure unsupervised training



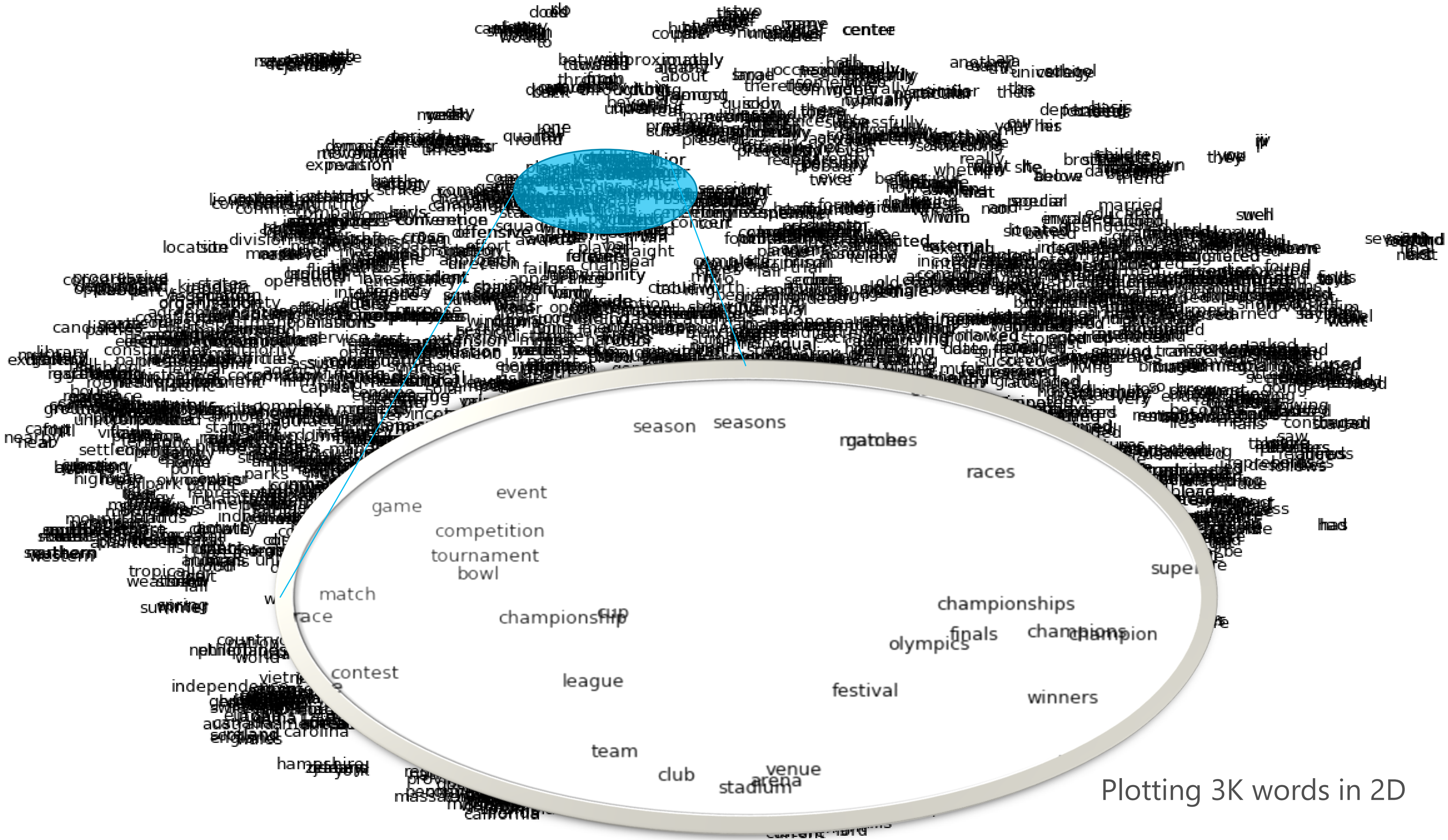
[Song et al. 2014]



Plotting 3K words in 2D



Plotting 3K words in 2D



Plotting 3K words in 2D

DSSM for Word Clustering and Analogy

Semantic clustering examples: top 3 neighbors of each word

king	earl (0.77)	pope (0.77)	lord (0.74)
woman	person (0.79)	girl (0.77)	man (0.76)
france	spain (0.94)	italy (0.93)	belgium (0.88)
rome	constantinople (0.81)	paris (0.79)	moscow (0.77)
winter	summer (0.83)	autumn (0.79)	spring (0.74)
rain	rainfall (0.76)	storm (0.73)	wet (0.72)
car	truck (0.8)	driver (0.73)	motorcycle (0.72)

Semantic analogy examples

$w_1 : w_2 = w_3 : ? \quad \Rightarrow \quad V_? = V_3 - V_1 + V_2$

summer : rain = winter : ?	snow (0.79)	rainfall (0.73)	wet (0.71)
italy : rome = france : ?	paris (0.78)	constantinople (0.74)	egypt (0.73)
man : eye = car : ?	motor (0.64)	brake (0.58)	overhead (0.58)
man : woman = king : ?	mary (0.70)	prince (0.70)	queen (0.68)
read : book = listen : ?	sequel (0.65)	tale (0.63)	song (0.60)

Broad impact on key text processing tasks

Semantic similarity modeling is critical in many text processing tasks

Deep Semantic Similarity Model (DSSM)

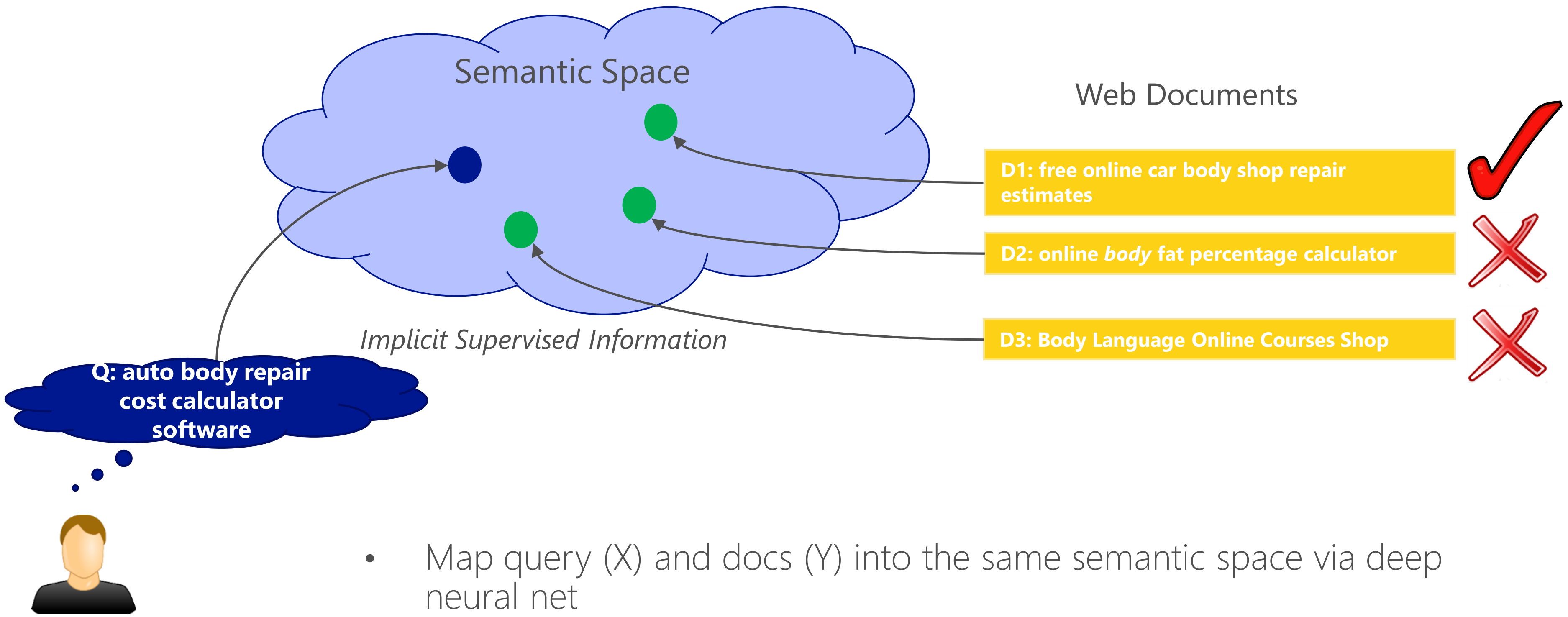
Compute semantic similarity between two text strings X and Y

- Map X and Y to feature vectors in a latent semantic space via deep neural net
- Compute the cosine similarity between the feature vectors

DSSM for ranking tasks

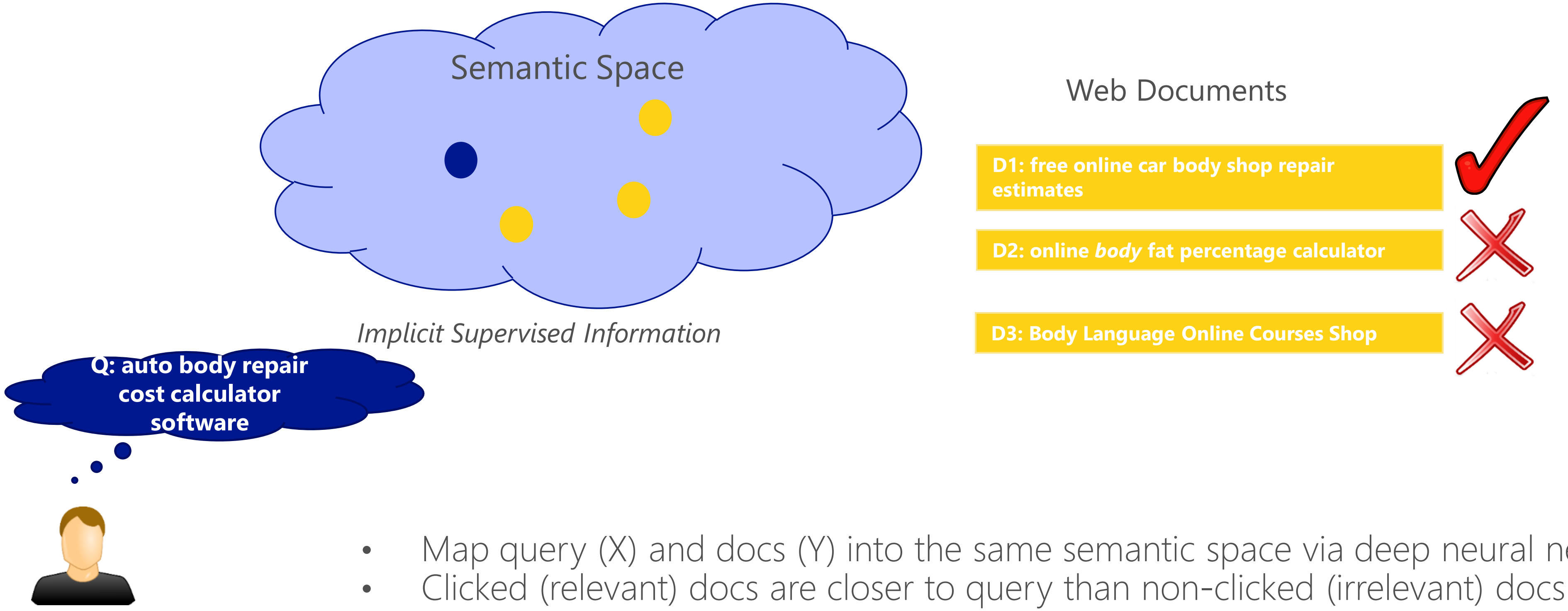
Tasks	***X***	***Y***
Web search	***Search query***	***Web documents***
Recommendation	***Doc in reading***	***Interesting things in doc or other docs***
Machine translation	***Sentence in language A***	***Translations in language B***

Learning DSSM on labeled X-Y pairs (clicked Q-D pairs)



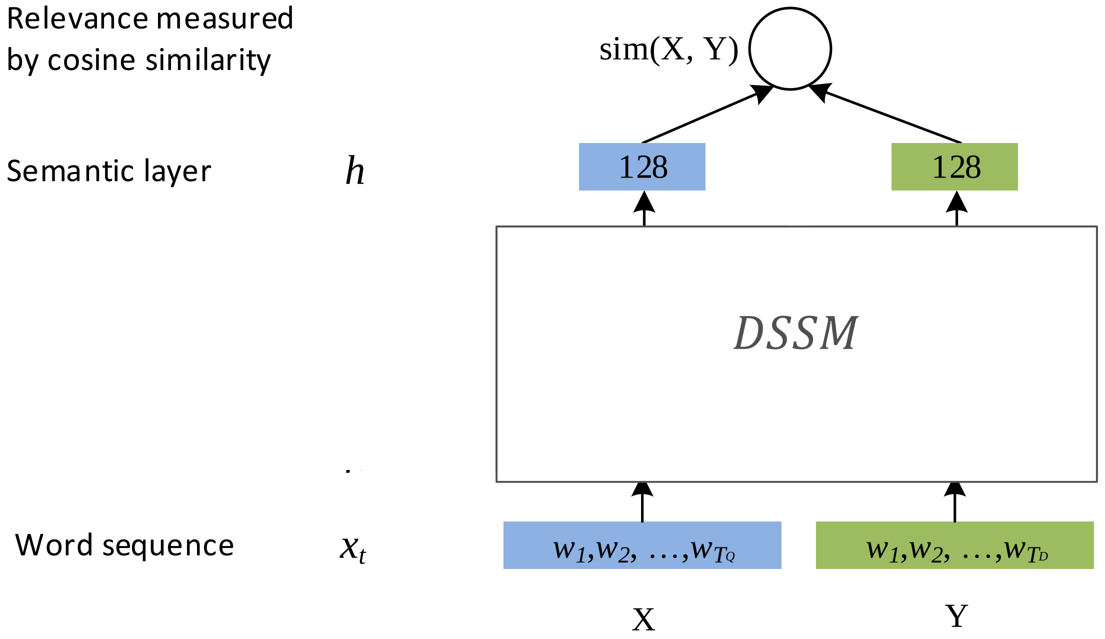
- Map query (X) and docs (Y) into the same semantic space via deep neural net

Learning DSSM on labeled X-Y pairs (clicked Q-D pairs)



- Map query (X) and docs (Y) into the same semantic space via deep neural net
- Clicked (relevant) docs are closer to query than non-clicked (irrelevant) docs in that space

DSSM: compute X-Y similarity in semantic space



Learning: maximize the similarity between relevant queries and docs

DSSM combines three pieces of MSR research

- DNN structure follows deep auto-encoder (Hinton and Deng 2009)
- The use of search logs for translation model training (Gao, He, Nie, 2010)
- Parameter optimization uses the pairwise rank loss based on cosine similarity (Yih et al. 2011; Gao et al. 2011)

https://microsoft.sharepoint.com/teams/DSSM_Text_Processing

Results on Web Search Ranking

#	Models	NDCG@1	Impr.	NDCG@3	Impr.
	Lexical Matching Models				
1	**BM25**	30.5		32.8	
2	**ULM [Zhai and Lafferty 2001]**	30.4	-0.1	32.7	-0.1
	Topic Models				
3	**PLSA [Hofmann 1999]**	30.5	+0.0	33.5	+0.7
4	**BLTM [Gao et al. 2011]**	31.6	+1.0	34.4	+1.6
	Clickthrough-based Translation Models				
5	**WTM [Gao et al. 2010]**	31.5	+1.0	34.2	+1.4
6	**PTM [Gao et al. 2010]**	31.9	+1.4	34.7	+1.9
	Deep Semantic Similarity Models				
7	**DSSM w/o convolutional layer**	32.0	+1.5	35.5	+2.7
8	**DSSM**	**34.2**	**+3.7**	**37.4**	**+4.6**

DSSM is the new state-of-the-art

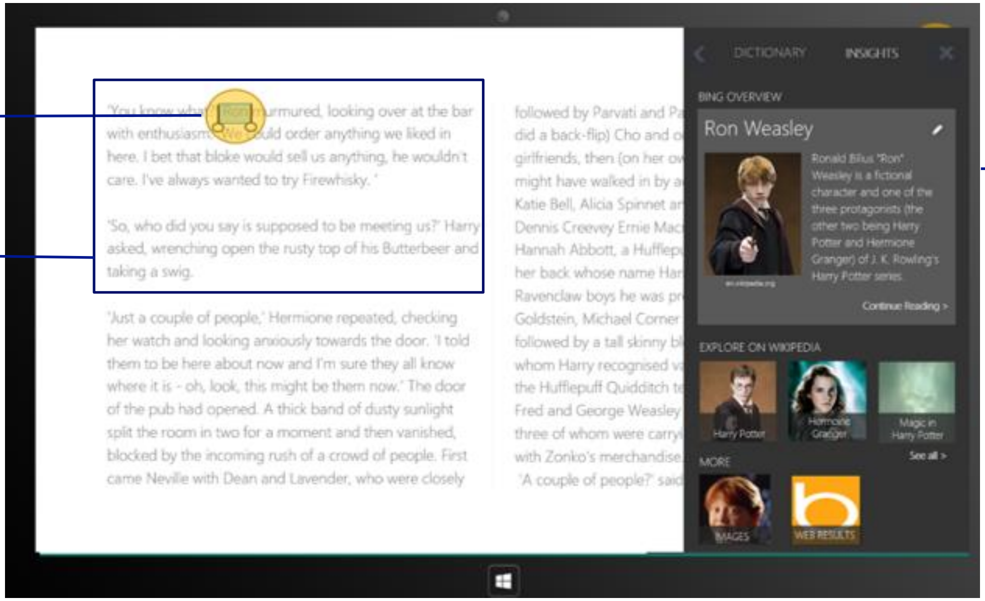
Modeling interestingness with DSSM

- Contextual entity search
  - Given a user-highlighted text span representing an entity of interest
  - Search for supplementary document for the entity
- Automatic highlighting
  - Given a document a user is reading
  - Discover the concepts/entities/topics that interest the user and highlight the corresponding text span
- Document prefetching
  - Given a document a user is reading
  -

DSSM for contextual entity ranking

Mention

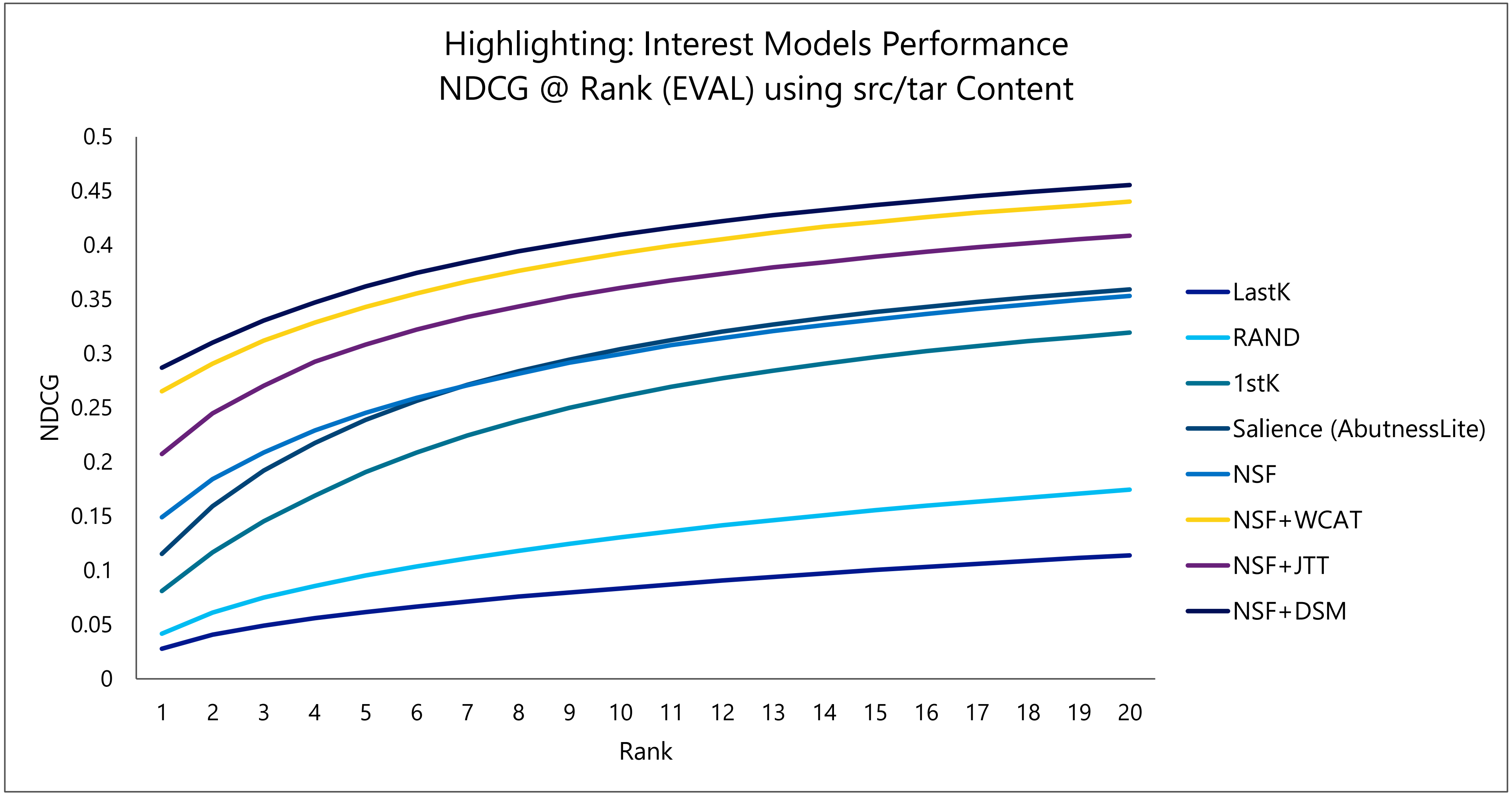
Context



KB Entity (reference doc)

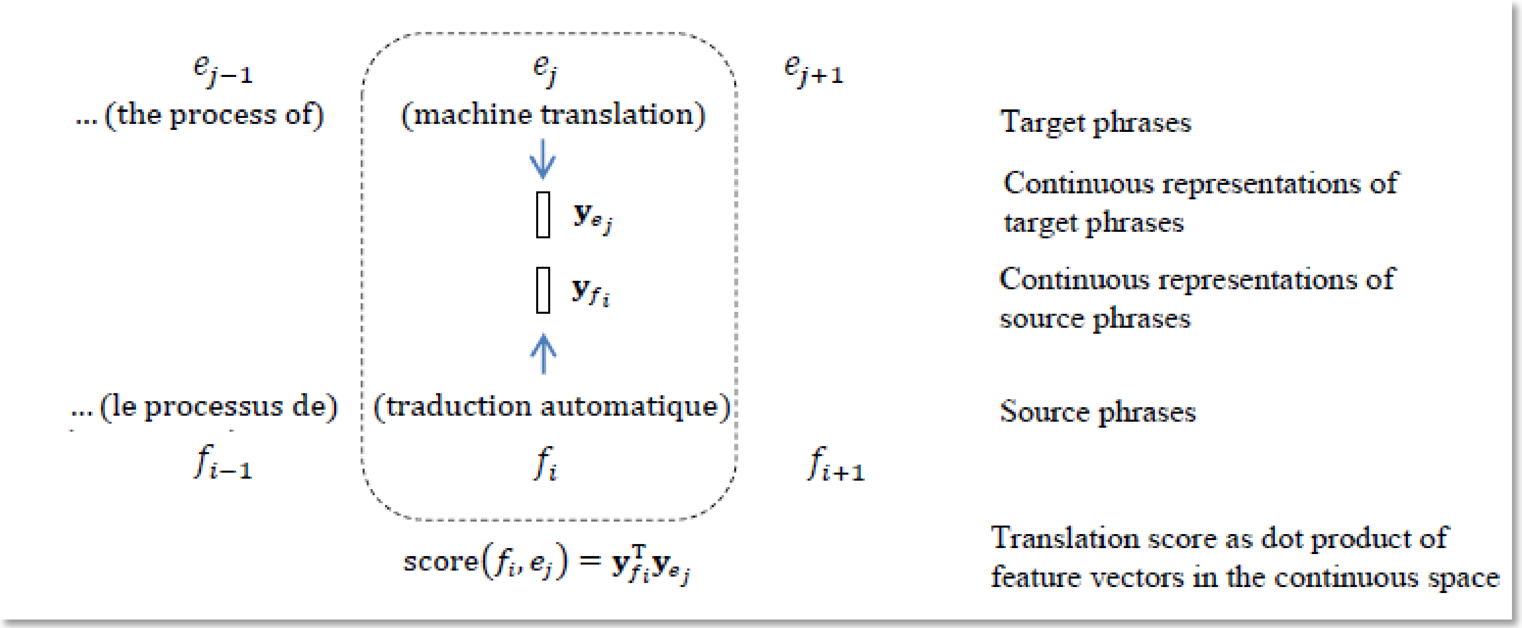
Ranker	AUC
BM25 (mention)	60%
Ranker (2306 features)	72%
DSSM (1 feature)	72%
Ranker+ DSSM	77%

- DSSM beats manually crafted text features
- +5 AUC gain over full ranker



- Features
  - DSM: DSSM
  - WCAT: semantic labels (page categories) assigned by editors
  - JTT: LDA-style topic models
  - NSF: non-semantic features
- DSSM learned features outperform the thousands of features coming from manually assigned labels (WCAT)

Results on Machine Translation



- Map the sentences in source/target languages into the same, language-independent semantic space
- The semantic translation model leads up to 1.3 BLEU improvement

[Gao, He, Yih, Deng, 2014]

DSSM: learning semantic similarity between *X* and *Y*

Tasks	***X***	***Y***
Web search	***Search query***	***Web documents***
Ad selection	***Search query***	***Ad keywords***
Entity ranking	***Mention (highlighted)***	***Entities***
Recommendation	***Doc in reading***	***Interesting things in doc or other docs***
Machine translation	***Sentence in language A***	***Translations in language B***
Nature User Interface	*Command (text/speech)*	*Action*
Summarization	*Document*	*Summary*
Query rewriting	*Query*	*Rewrite*
Image retrieval	*Text string*	*Images*
…	…	…

[Huang et al. 2013; Shen et al. 2014; Gao et al. 2014a; Gao et al. 2014b]

Save the planet and return your name badge before you leave (on Tuesday)

Microsoft