KG4IR '17
Tokyo, Japan
Aug. 11, 2017

Deep Approaches to Semantic Text Matching

Jun Xu

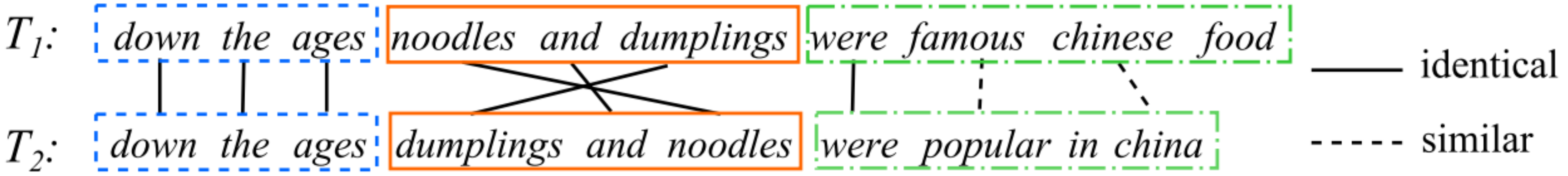
Institute of Computing Technology,
Chinese Academy of Sciences

中国科学院计算技术研究所
Institute of Computing Technology,Chinese Academy of Sciences

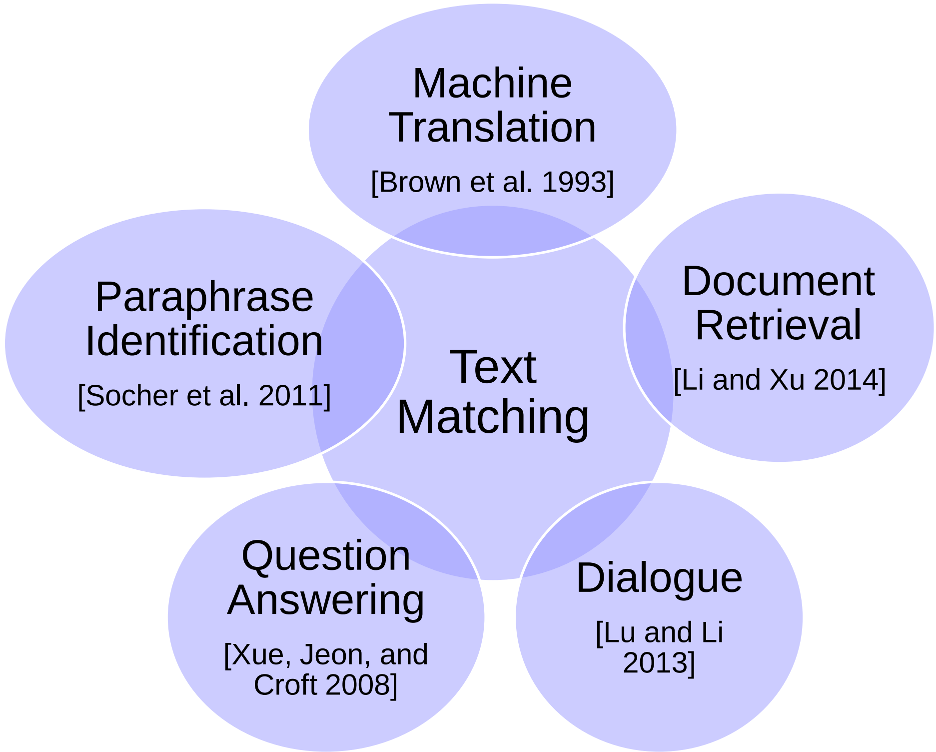
Outline

- Semantic text matching is important
- Word representation: bridging the semantic gap
- Sentence matching: capturing the proximity
- Summary

Semantic Text Matching



Are these two sentences **similar**?



Accuracies of Natural Language Analysis

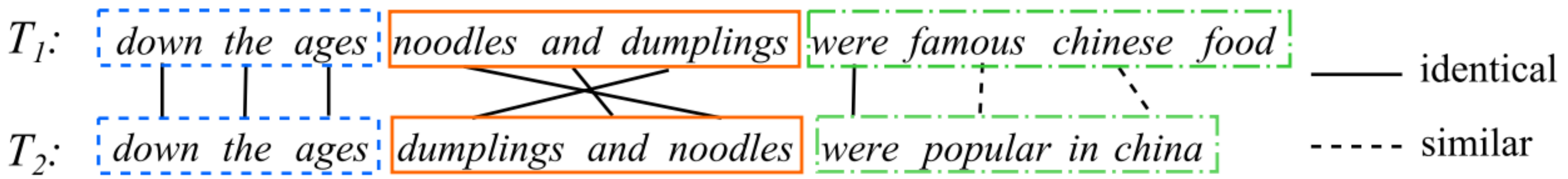
- Lexical Analysis (word segmentation and part-of-speech tagging): practically usable
- Syntactic Analysis: almost usable
- Semantic Analysis: still difficult
- Programmatic Analysis: ?

	English	Chinese
Prgrammatic Analysis	?	?
Semantic Role Labeling	>=87%	>=75%
Syntactic Analysis	>=90%	>=80%
Part of Speech Tagging	>=97%	>=93%
Word Segmentation	NA	>=95%

Slides from Hang Li

Current Approach:
Avoid Understanding and Conduct **Matching**

Text semantic matching challenges



- Word level: semantic gaps between words
  - Two words has similar meanings
  - “popular” ~ “famous”; “china” ~ “chinese"
- Sentence level: proximity matching between sentences
  - The matching positions do matter
  - “noodles and dumpling” – “dumplings and noodles”
- Need to consider them simultaneously

Learning to (semantic) text match

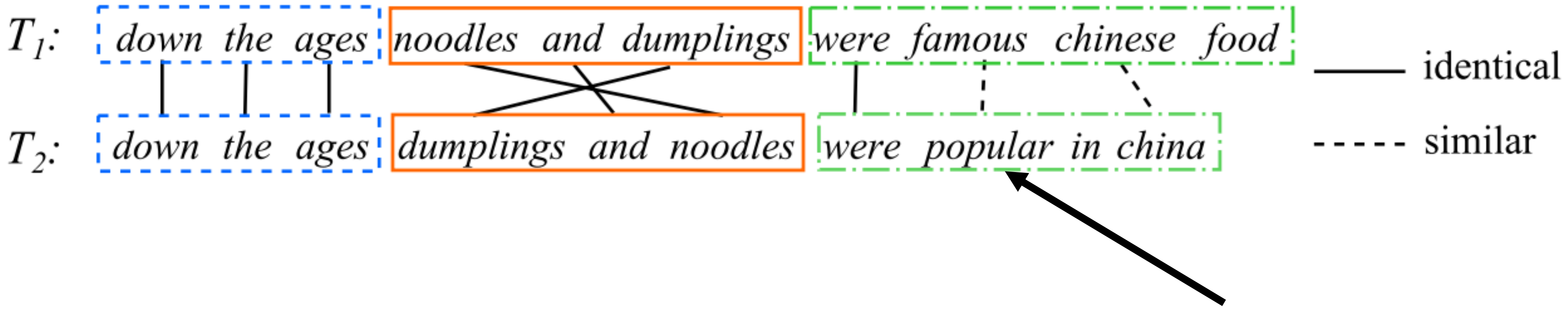
- The problem can be formulized as

$$\text{Match}(T_1, T_2) = F\big(\phi(T_1), \phi(T_2)\big)$$

  - ϕ: mapping text to representation vector
  - F: scoring function based on representation
- Learning the model parameters
  - Learning the representation ϕ
  - Learning the scoring function F

Outline

- Matching is important for text analysis
- Word representation: bridging the semantic gap
- Sentence matching: capturing the proximity
- Summary



How similar "popular" to "famous"?

Local representation of words

- Words are the building blocks of texts
- NLP treats words mainly (rule-based / statistical approach at least) as atomic symbols:

  Man | Woman | Dog | Computer

- also known as "one-hot" or local representation

One-Hot Representation	
man	[1,0,...,0,0,...,0,0]
woman	[0,1,...,0,0,...,0,0]
dog	[0,0,...,1,0,...,0,0]
computer	[0,0,...,0,0,...,1,0]

- local representation: each word is locally represented by a distinct node.

Limitation of local representations

- Local representation makes a strong independent assumption between words

Local Representation	
man	[1,0,...,0,0,...,0,0]
woman	[0,1,...,0,0,...,0,0]
car	[0,0,...,1,0,...,0,0]
automobile	[0,0,...,0,0,...,1,0]

cos(car, automobile) = 0!

cos(man, women) = cos(man, car)

- Local representation is not efficient
  - require N nodes to represent N words

The distributional hypothesis
[Harris, 1954, Firth, 1957]

"Words that occur in the same contexts tend to have similar meanings."

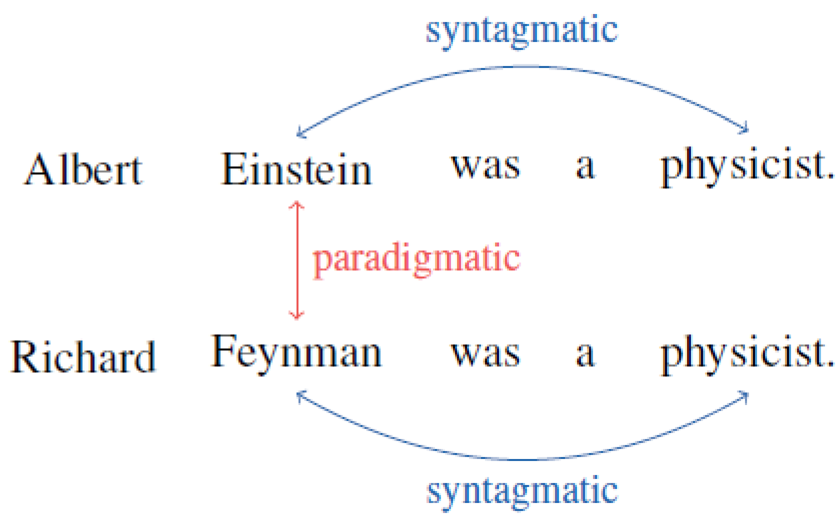
–Zellig Harris [Harris, 1954]

"You shall know a word by the company it keeps."

—J.R. Firth

- Discover semantic from **external** information
  - A word is just an ID, its meaning depends on other words (company it keeps, or context)
- One Hypothesis, two interpretations

Two interpretations: Syntagmatic and paradigmatic [Sahlgren, 2008]



- Syntagmatic: words co-occur in the same text region (they are related)
- Paradigmatic: words occur in the same context, may not at the same time (they are similar)

Modeling syntagmatic relation

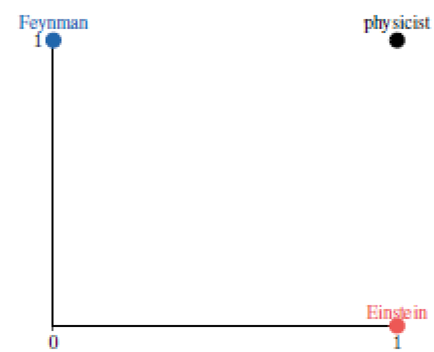
syntagmatic

Albert Einstein was a physicist.

Richard Feynman was a physicist.

syntagmatic

	d_1	d_2
Einstein	1	0
Feynman	0	1
physicist	1	1



LSI, LDA, PV-DBOW $\cdots$

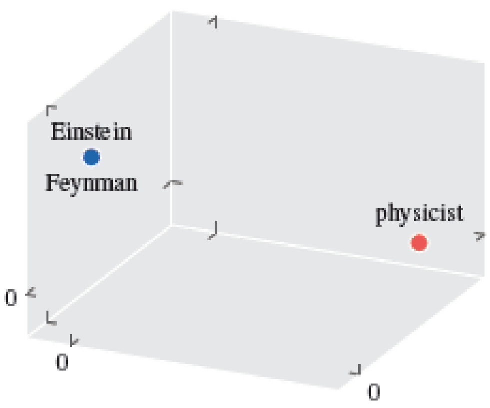
Modeling paradigmatic relation

Albert **Einstein** was a physicist.

↕ paradigmatic

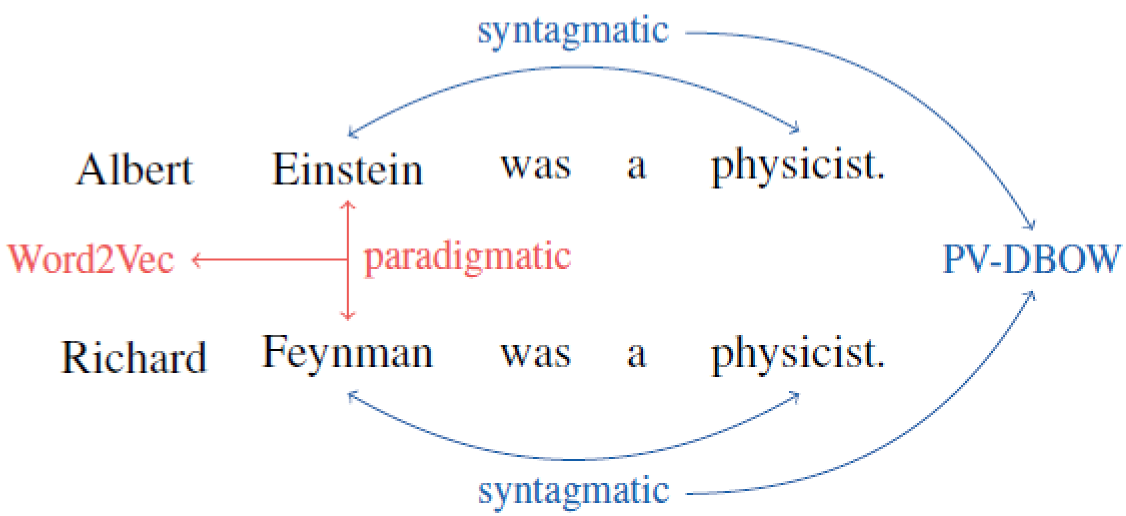
Richard **Feynman** was a physicist.

	Einstein	Feynman	physicist
Einstein	0	0	1
Feynman	0	0	1
physicist	1	1	0



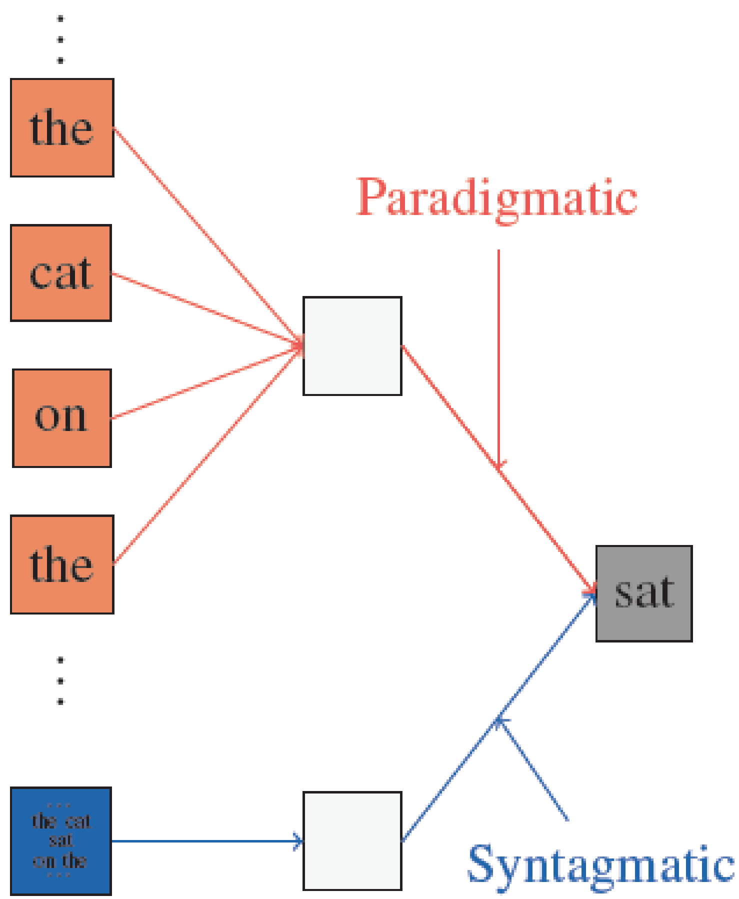
NLMs, Word2Vec, GloVe · · ·

Modeling them jointly



Sun et al., Learning Word Representations by Jointly Modeling Syntagmatic and Paradigmatic Relations. In Proc. ACL 2015.

Parallel document content model (PDC)



$$\ell = \sum_{n=1}^{N} \sum_{w_i^n \in d_n} \left(\log p(w_i^n | h_i^n) + \log p(w_i^n | d_n) \right)$$

Negative Sampling

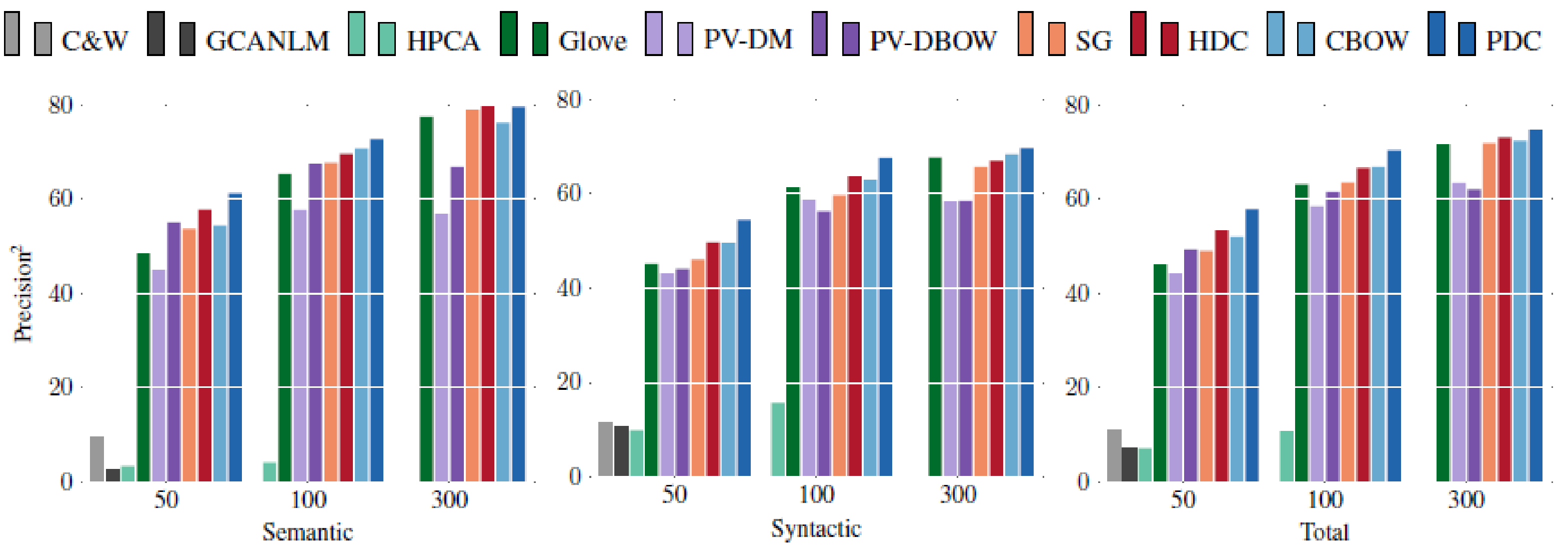
$$\ell = \sum_{n=1}^{N} \sum_{w_i^n \in d_n} \Big(\log \sigma(\vec{w_i^n} \cdot \vec{h_i^n}) + \log \sigma(\vec{w_i^n} \cdot \vec{d_n}) + k \cdot \mathbb{E}_{w' \sim P_{\mathrm{nw}}} \log \sigma(-\vec{w'} \cdot \vec{h_i^n}) + k \cdot \mathbb{E}_{w' \sim P_{\mathrm{nw}}} \log \sigma(-\vec{w'} \cdot \vec{d_n}) \Big)$$

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

PDC	PV-DM
MF for W-D + W-C [Levy and Goldberg, 2014]	not clear

Empirical evaluation: word analogy

- Google test set [Mikolov et al., 2013]
  - Semantic: "Beijing is to China as Paris is to __"
  - Syntactic: "big is to bigger as deep is to __"



Diversify the results

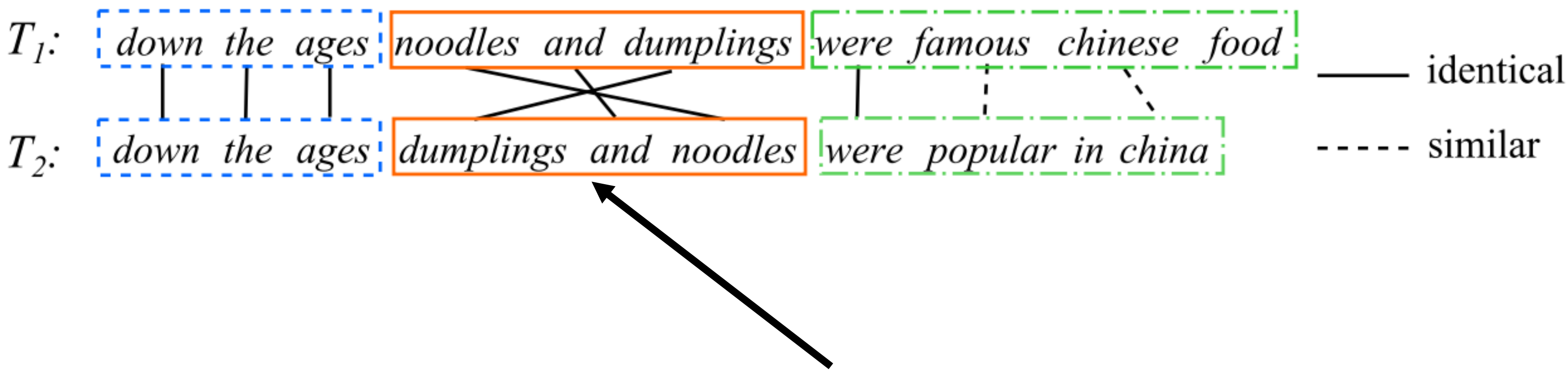
Top 5 similar words to **Feynman**

CBOW	SG	PDC	HDC	PV-DBOW
einstein	schwinger	geometrodynamics	schwinger	physicists
schwinger	quantum	bethe	electrodynamics	spacetime
bohm	bethe	semiclassical	bethe	geometrodynamics
bethe	einstein	schwinger	semiclassical	tachyons
relativity	semiclassical	perturbative	quantum	einstein

Paradigmatic　　Syntagmatic

Outline

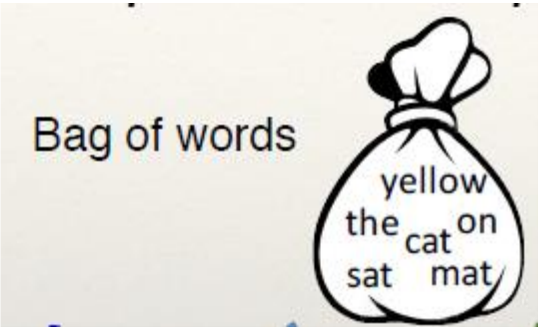
- Matching is important for text analysis
- Word representation: bridging the semantic gap
- Sentence matching: capturing the proximity
- Summary

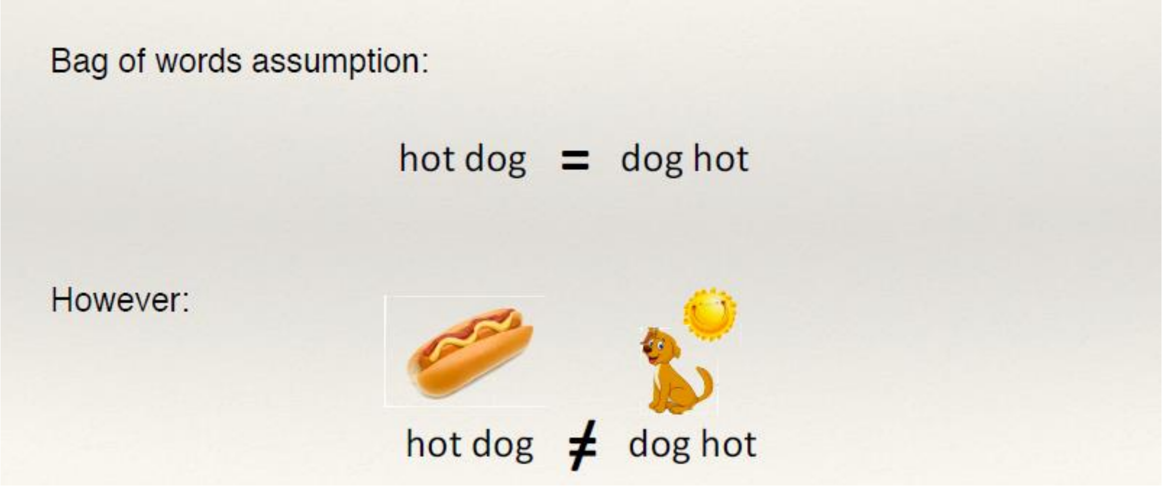


How similar "noodles and dumplings" to "dumplings and noodles"?

Bag of words

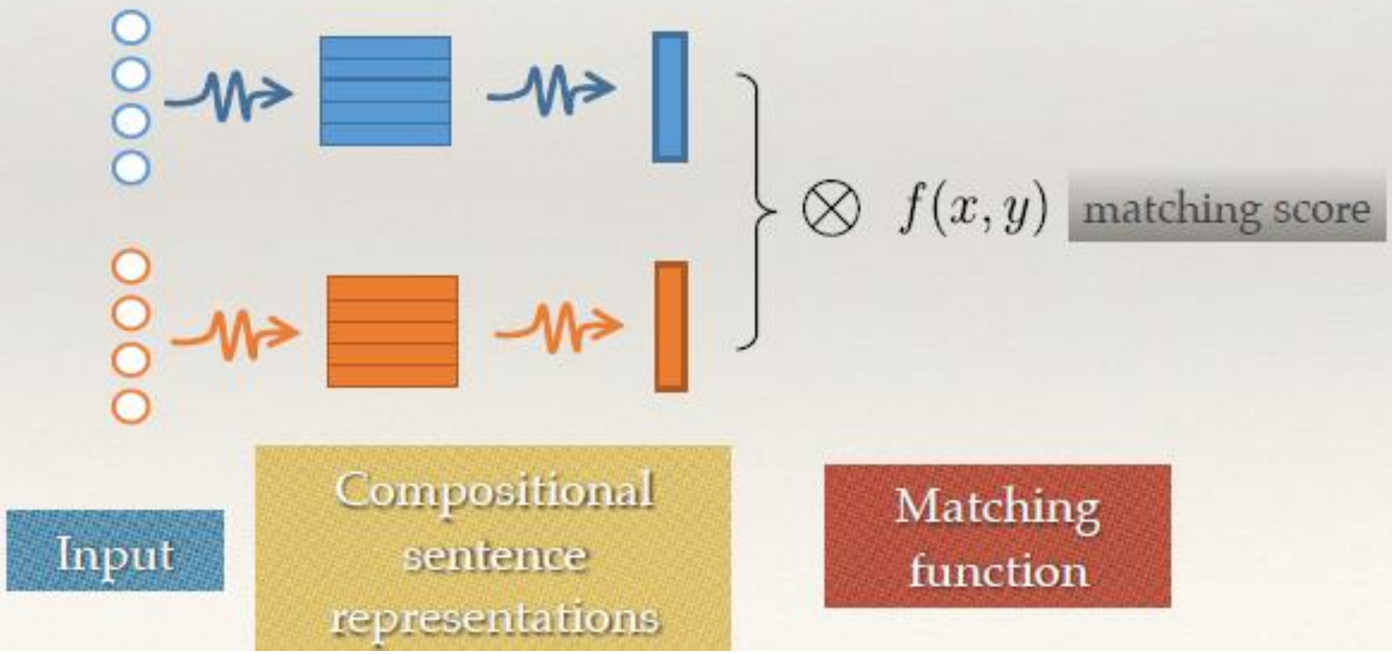
- Bag of words representation of sentences
  - the yellow cat sat on the mat
  - the cat sat on the yellow mat
- Heuristic matching function
  - Cosine similarity, BM25 ……
- However, order of words is important



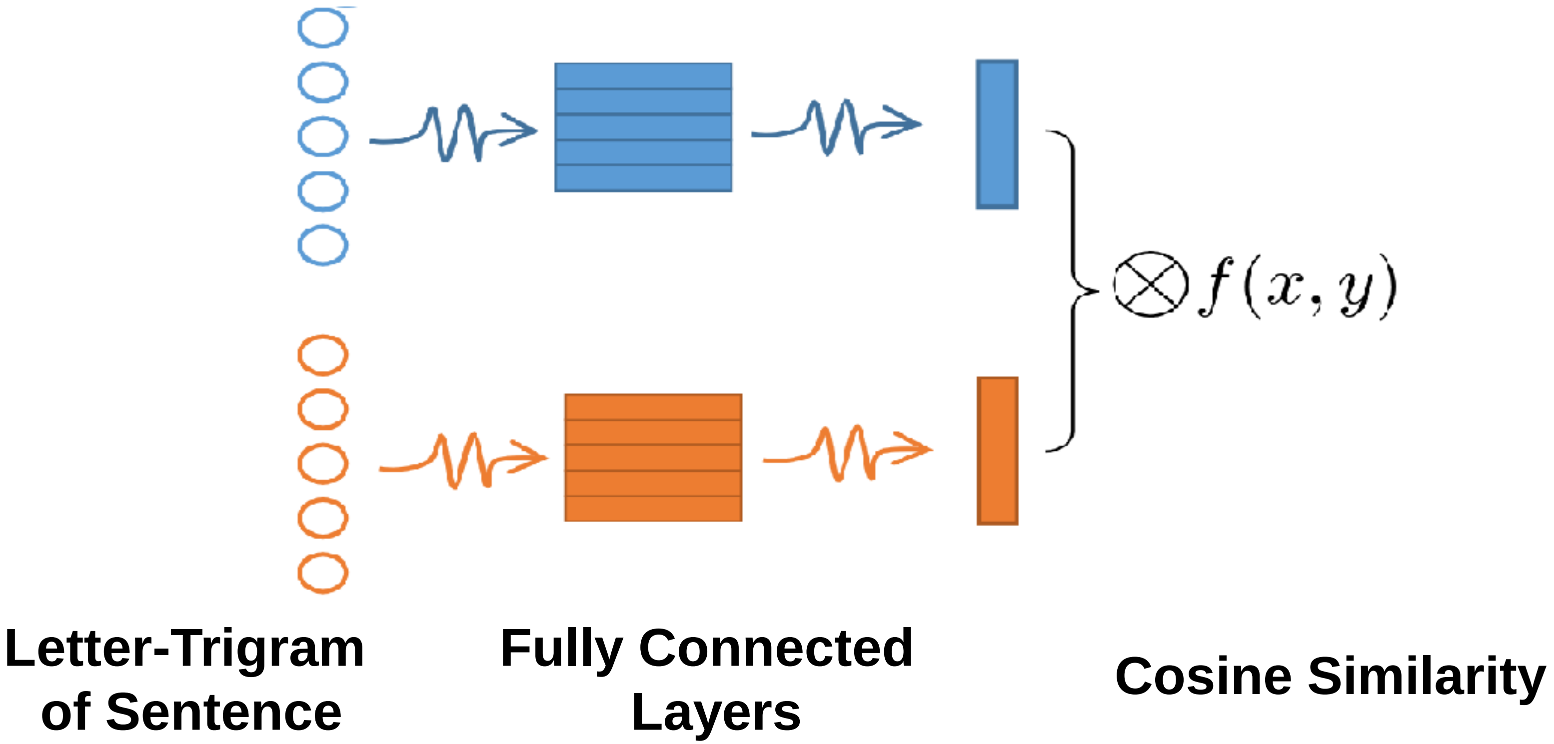


Approach 1: composition focused

- Step 1: Composite sentence representation $\phi(x)$
- Step 2: Matching between the representations $F(\phi(x), \phi(y))$



Composition focused methods example: DSSM



Huang P-S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data//Proceedings of the 22nd ACM international conference on CIKM, 2013: 2333-2338.

DSSM Input – letter-trigram

- Word One-Hot Representation

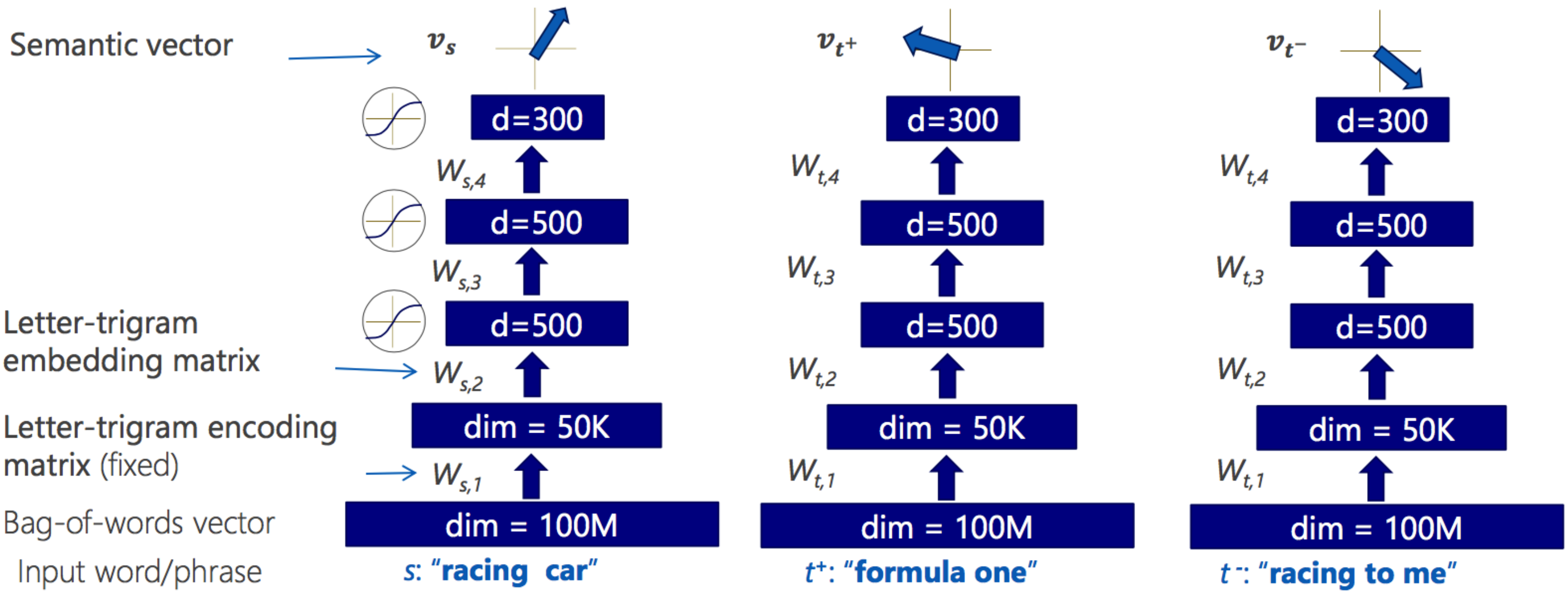
  Candy [o o o o o **1** o o o o o o o o o o o ...]

  Store [o o o o o o o o o o o o o o o **1** o ...]

- Letter-Trigram Representation
  - #candy# | #store#   can split into:
  - #ca | can | and | ndy | dy# | #st | sto | tor | ore | re#
  - [o o 1 o o ... o 1 o 1 ... o o ...]

- Compact representation: |words| (500K) -> |letter-trigrams| (30K)
- Generalize to unseen words
- Robust to misspelling, inflection, etc

DSSM - Composite Embedding



DSSM - Aggregate Matching Score

- Compute Cosine similarity between semantic vectors

$$S = \frac{x^T \cdot y}{|x| \cdot |y|}$$

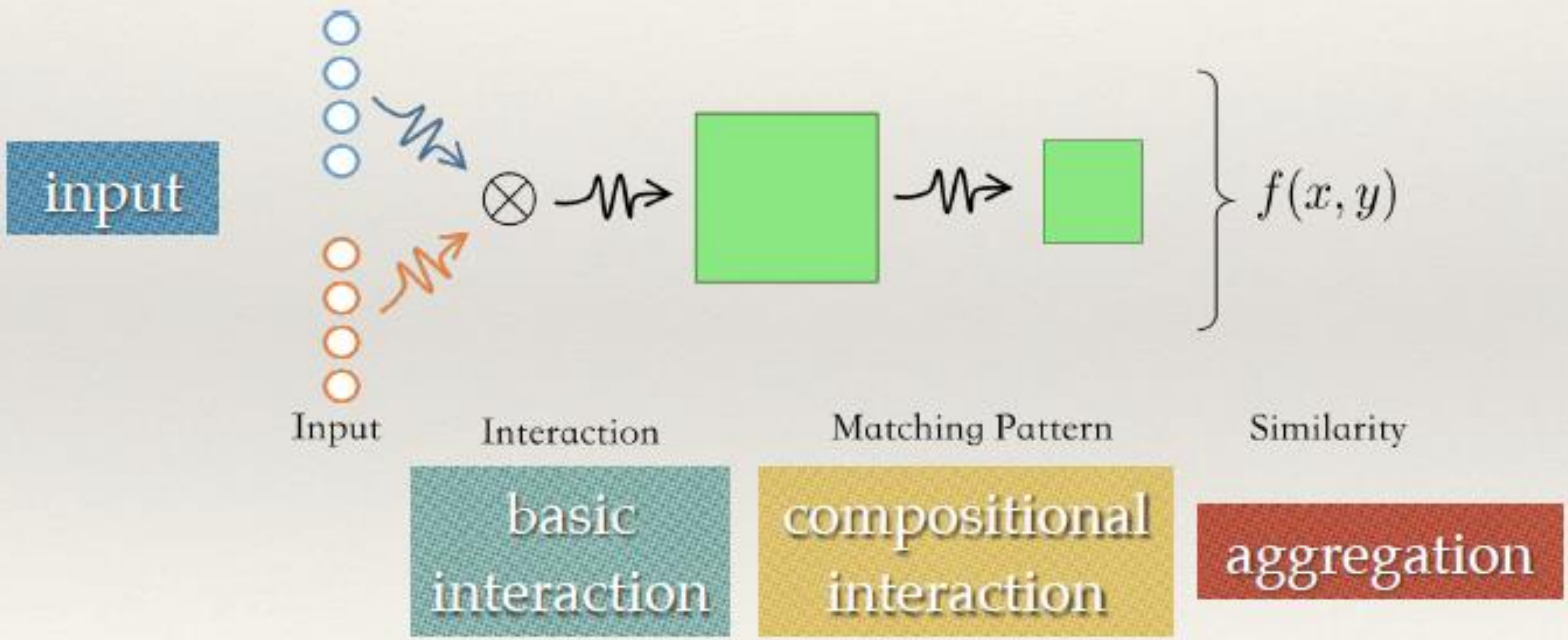
- Training
  - A query q and a list of docs $D = \{d^+, d_1^-, \dots, d_k^-\}$
  - d^+positive doc, $d_1^-, \dots, d_k^-$ negative docs to q
  - Objective:

$$P(d^+|q) = \frac{\exp(\gamma\, cos(q, d^+))}{\sum_{d \in D} \exp(\gamma\, cos(q, d))}$$

  - Optimize to maximize $P(d^+|q)$. SGD Method.

Approach 2: interaction focused

- Step 1: Construct basic low-level interaction signals
- Step 2: Aggregate matching patterns

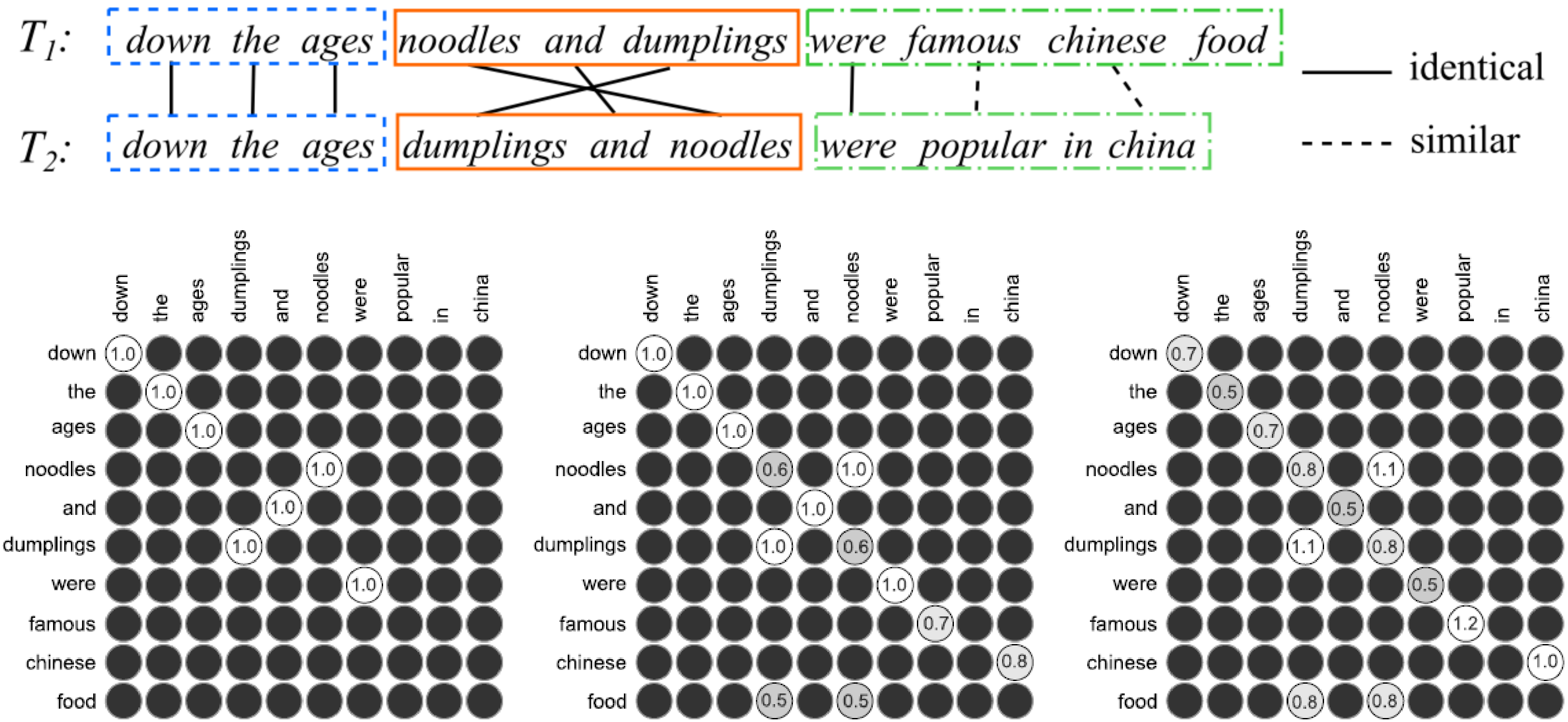


Interaction focused methods example: MatchPyramid

- Challenges
  - Representation: representing the word level **matching signals** as well as the **matching positions**
  - Modeling: discovering the **matching patterns** between two texts
- Our solutions
  - Step 1: representing as matching matrix
  - Step 2: matching as image recognition

Pang et al., Text Matching as Image Recognition. In Proc. AAAI 2016.

Step 1: matching matrix



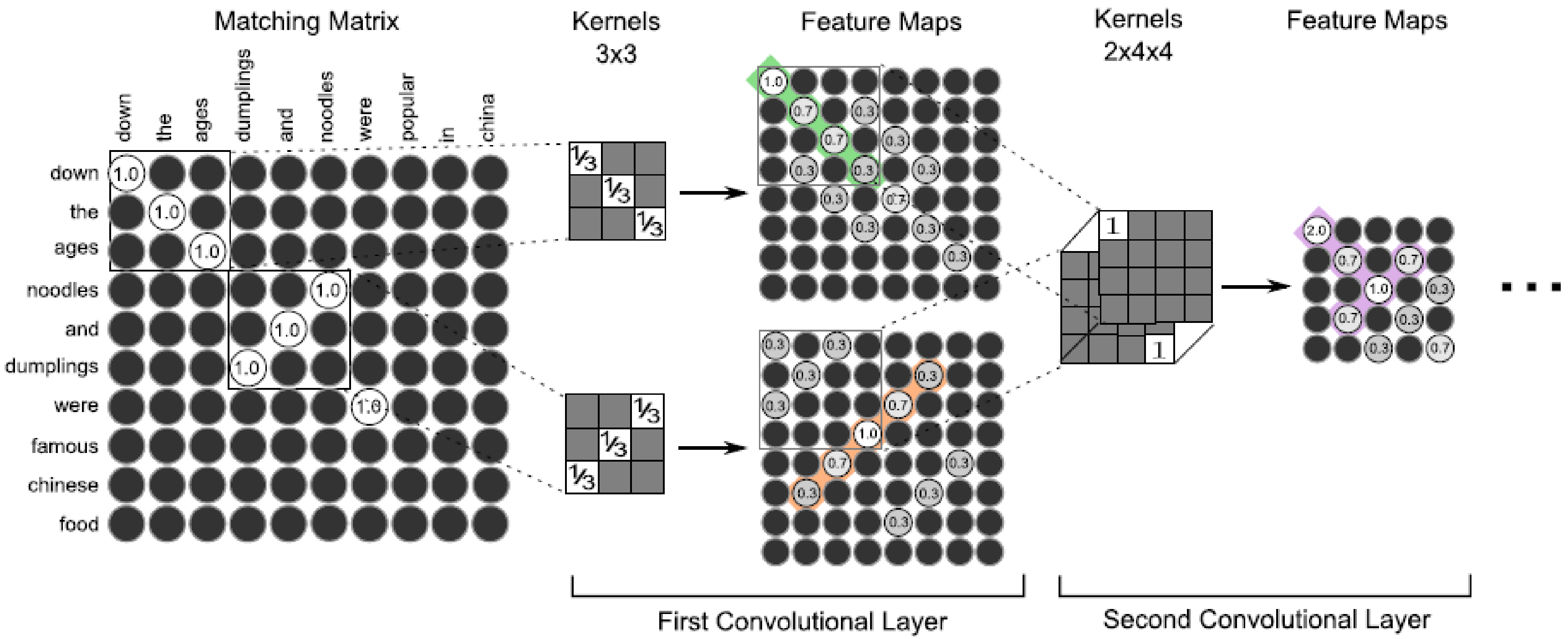
(a) Matching Matrix-Indicator (b) Matching Matrix-Cosine (c) Matching Matrix-Dot Product

$$\mathbf{M}_{ij} = \mathbb{I}_{\{w_i = v_j\}} = \begin{cases} 1, & \text{if } w_i = v_j \\ 0, & \text{otherwise.} \end{cases}$$

$$\mathbf{M}_{ij} = \frac{\vec{\alpha_i}^\top \vec{\beta_j}}{\|\vec{\alpha_i}\| \cdot \|\vec{\beta_j}\|}$$

$$\mathbf{M}_{ij} = \vec{\alpha_i}^\top \vec{\beta_j}.$$

Step 2: matching as image recognition



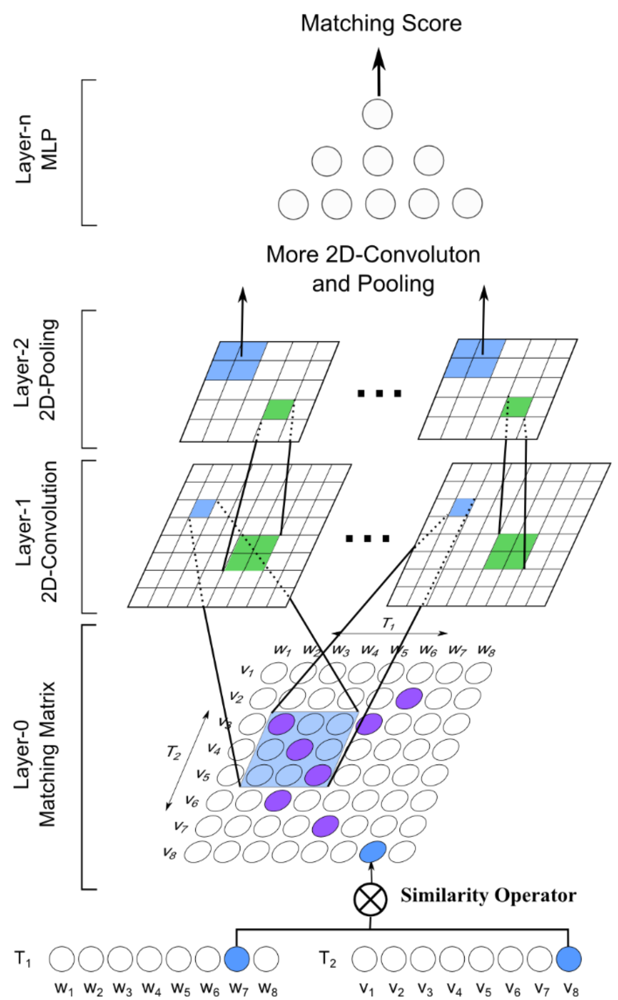
Putting together: MatchPyramid

Hierarchical Convolution

Capturing rich matching patterns

Matching Matrix

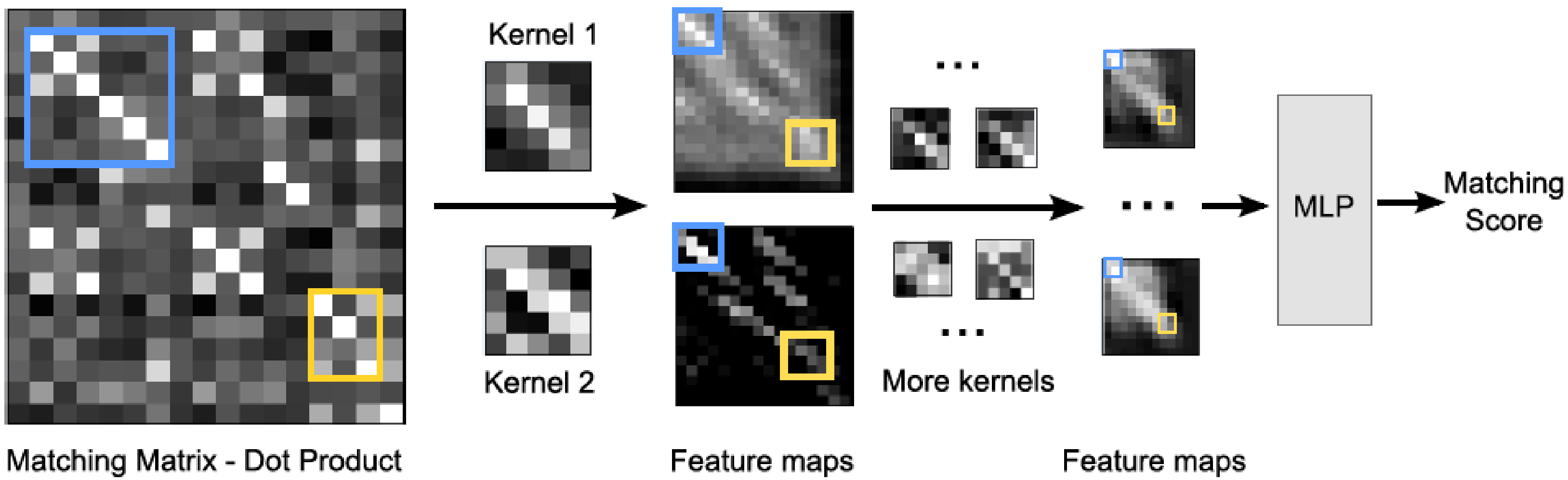
Bridging the semantic gap between words



MatchPyramid discovers text matching patterns

T_1: PCCW's chief operating officer, Mike Butcher, and Alex Arena, the chief financial officer, will report directly to Mr So.

T_2: Current Chief Operating Officer Mike Butcher and Group Chief Financial Officer Alex Arena will report to So.



Empirical evaluation: Paraphrase Identification (MSRP)

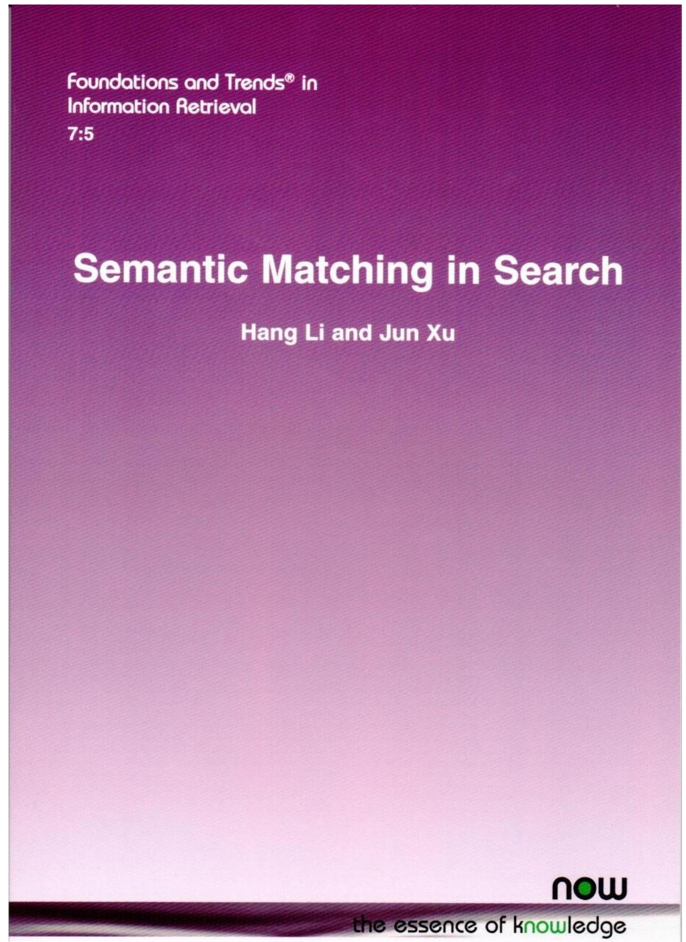
	Model	Accuracy(%)	F1(%)
Traditional	TF-IDF	70.31	77.62
Composition Focused	DSSM	70.09	80.96
	CDSSM	69.80	80.42
	ARC-I	69.60	80.27
	uRAE	76.80	83.60
	MultiGranCNN	78.10	84.40
	MV-LSTM	75.40	82.80
Interaction Focused	ARC-II	69.90	80.91
	MatchPyramid	75.94	83.01
	Match-SRNN	74.50	81.70

Outline

- Matching is important for text analysis
- Word representation: bridging the semantic gap
- Sentence matching: capturing the proximity
- Summary

Summary

- Semantic matching in text is fundamental for QA, IR, and paraphrasing etc.
- Semantic matching is challenging
  - Semantic gaps between words
  - Proximity matching between sentences
- Our solutions
  - Semantic: distributed word representation with external (content) information
  - Proximity: Composition focused and interaction focused methods, e.g., MatchPyramid



http://www.bigdatalab.ac.cn/~junxu/publications/SemanticMatchingInSearch_2014.pdf
http://www.nowpublishers.com/articles/foundations-and-trends-in-information-retrieval/INR-035

Thank you!

Q&A

junxu@ict.ac.cn

http://www.bigdatalab.ac.cn/~junxu