A Study of MatchPyramid Models on Ad-hoc Retrieval

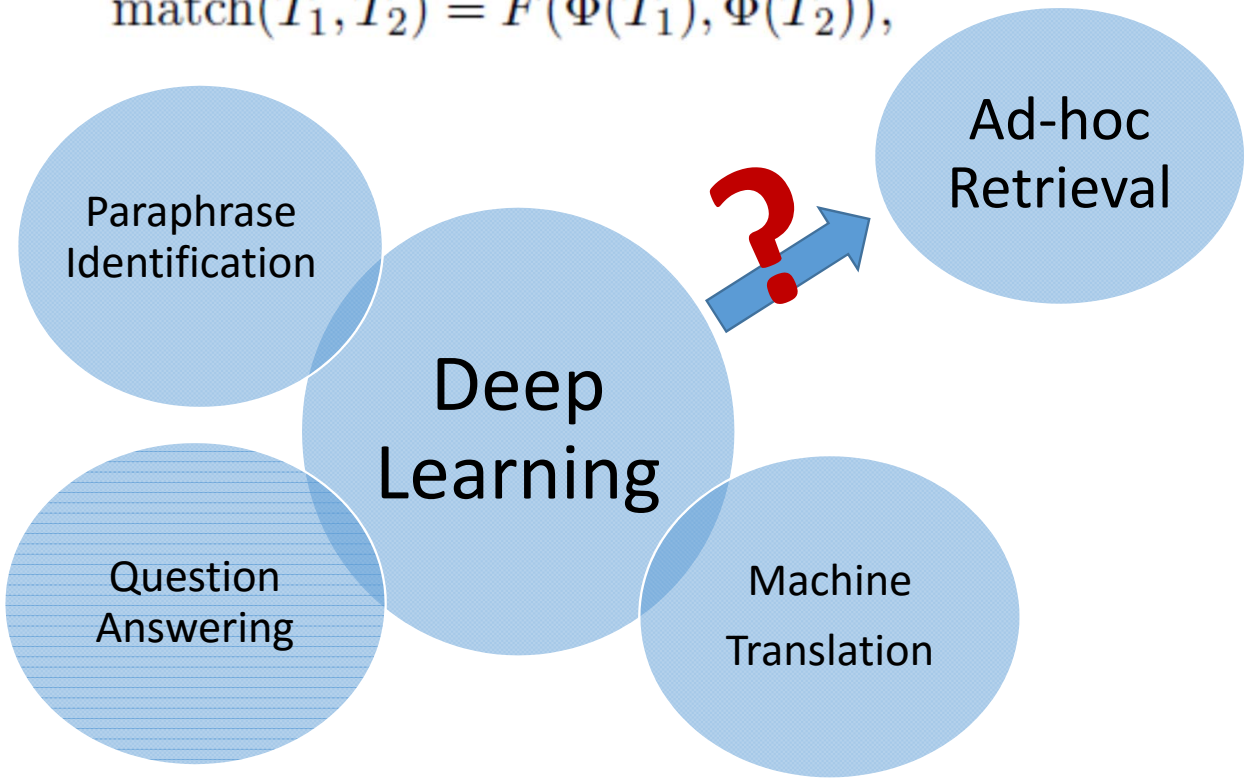
Liang Pang, Yanyan Lan, Jiafeng Guo,
Jun Xu, Xueqi Cheng

Institute of Computing Technology, Chinese Academy of Sciences

Text Matching

Many text based applications have been formalized as a matching task

$$\text{match}(T_1, T_2) = F(\Phi(T_1), \Phi(T_2)),$$



Lack the evaluation of deep matching models on the standard ad-hoc retrieval tasks

A Representative model： MatchPyramid

- A State-of-the-art text matching model [AAAI 2016]

Table 1: Results on MSRP.

Model	Acc.(%)	F_1(%)
ALLPOSITIVE	66.50	79.87
TF-IDF	70.31	77.62
DSSM	70.09	80.96
CDSSM	69.80	80.42
ARC-I	69.60	80.27
ARC-II	69.90	80.91
MP-IND	75.77	82.66
MP-COS	75.13	82.45
MP-DOT	**75.94**	**83.01**

Paraphrase Identification

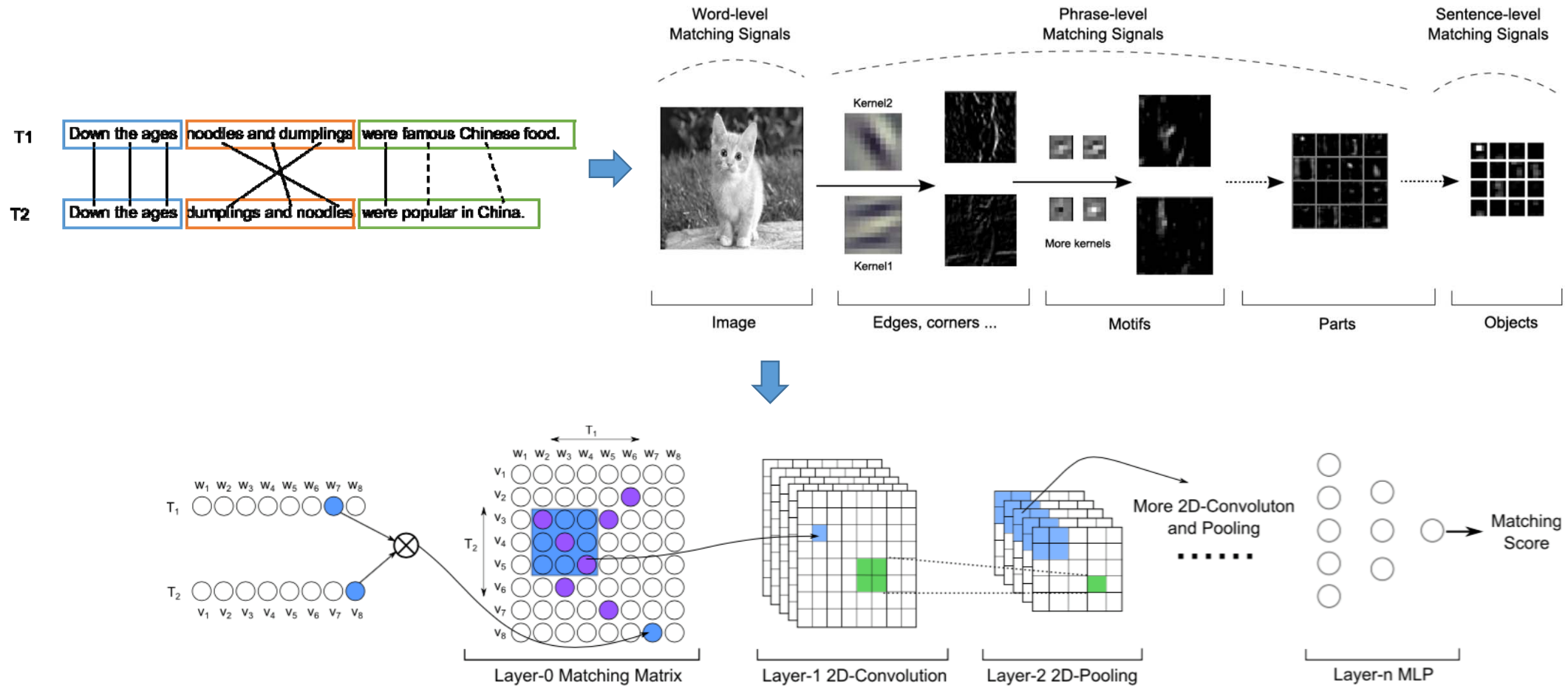
Table 2: Results on the task of paper citation matching.

Model	Acc.(%)	F_1(%)
ALLPOSITIVE	33.33	50.00
TF-IDF	82.63	70.21
DSSM	71.97	29.88
CDSSM	69.84	19.97
ARC-I	84.51	76.79
ARC-II	86.48	79.57
MP-IND	73.76	44.71
MP-COS	86.65	79.70
MP-DOT	**88.73**	**82.86**

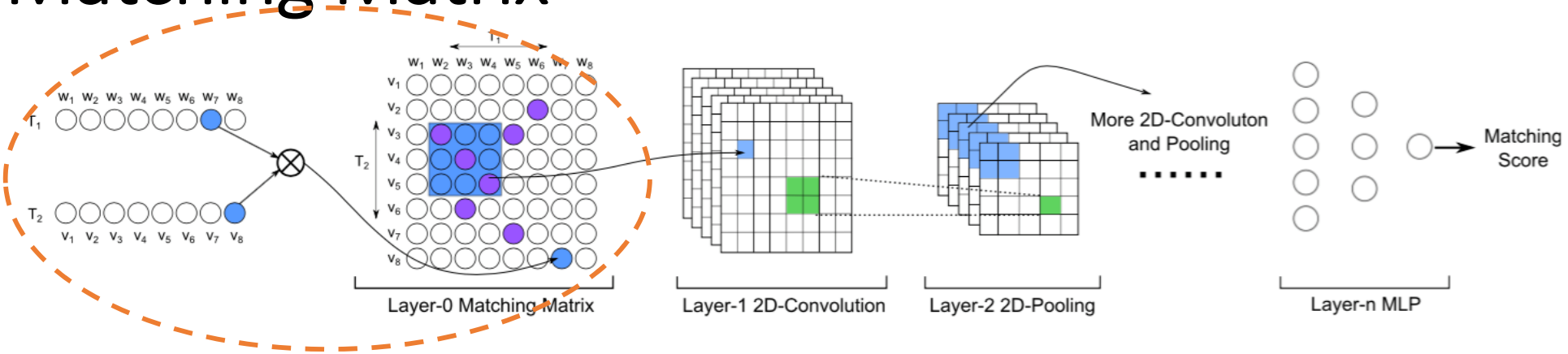
Paper Citation Matching

A Brief Introduction to MatchPyramid

Text Matching as Image Recognition



Matching Matrix



Capture the local interactions, and bridge the gap between text matching and image recognition

$$\mathbf{M}_{ij} = w_i \otimes v_j$$

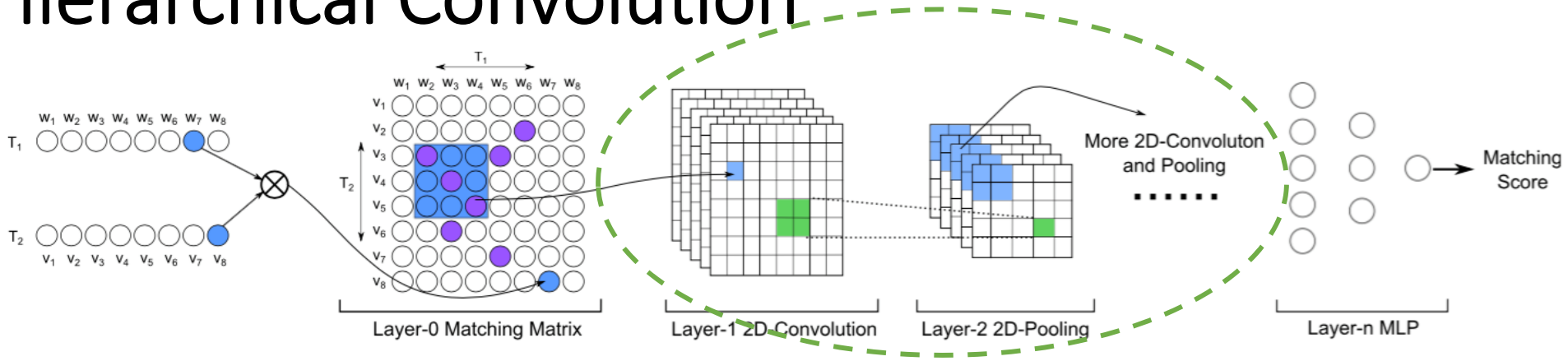
Indicator: $$\mathbf{M}_{ij} = \mathbb{I}_{\{w_i = v_j\}} = \begin{cases} 1, & \text{if } w_i = v_j \\ 0, & \text{otherwise.} \end{cases}$$

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

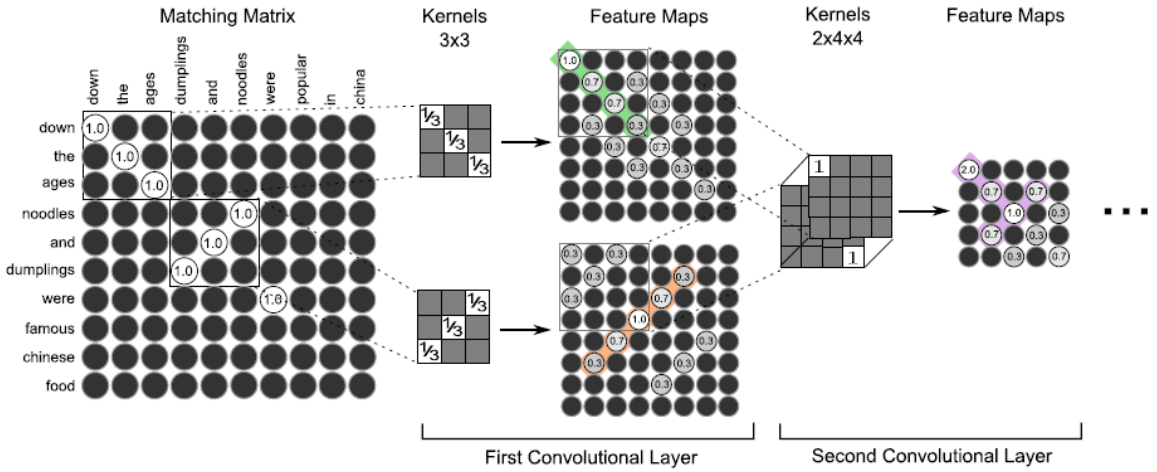
Cosine: $$\mathbf{M}_{ij} = \frac{\vec{\alpha_i}^\top \vec{\beta_j}}{\|\vec{\alpha_i}\| \cdot \|\vec{\beta_j}\|}$$

Gaussian Kernel: $$\mathbf{M}_{ij} = e^{-\|\vec{\alpha_i} - \vec{\beta_j}\|^2}$$

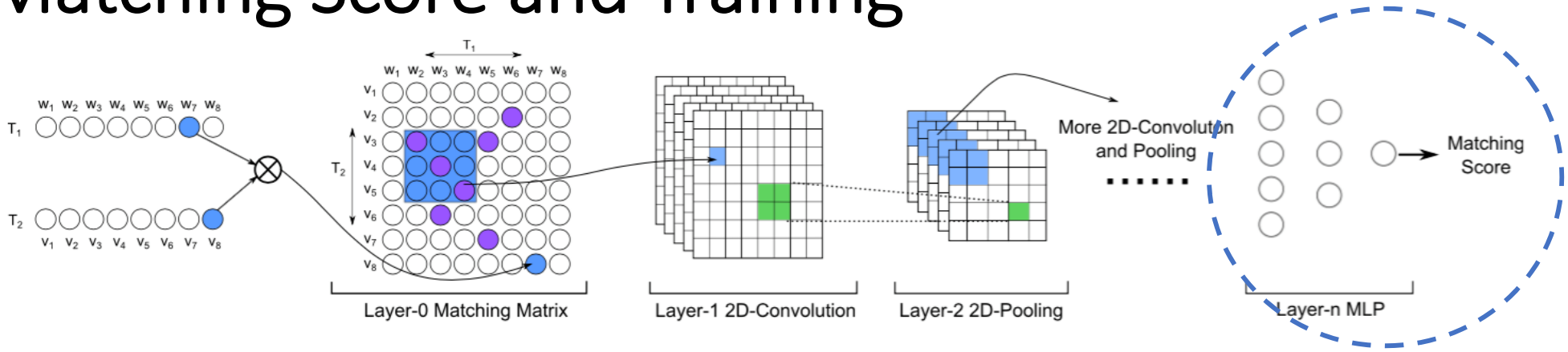
Hierarchical Convolution



A Way to Capture Rich Matching Patterns (e.g., n-grams, n-terms)



Matching Score and Training



Matching Score Generation

Multi-Layer Perception

$$(s_+, s_-)^\intercal = W_2\sigma(W_1 z + b_1) + b_2$$

Training

$$\mathcal{L}(s_+, s_-; \Theta) = \max(0, 1 - s_+ + s_-)$$

Adagrad [Duchi, Hazan,and Singer 2011]

Experiments on Ad-hoc Retrieval

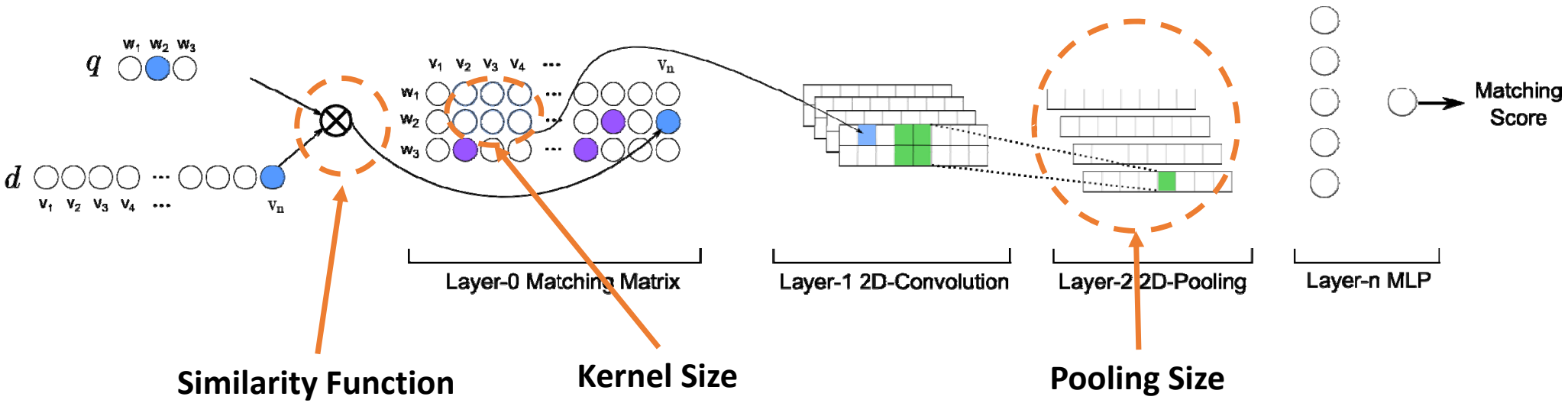
Experimental settings

- TREC Robust04
  - The topics are collected from TREC Robust Track 2004.
  - Use the Galago Search Engine in this experiment, and both queries and documents are white-space tokenized, lower-cased, and stemmed during indexing and retrieval.
  - Truncate the document length to 500.

Table 1: Statistics of the TREC collection Robust04.

#Vocab	#Doc	#Query	#Retrieval doc	Avg doc
0.6M	0.5M	250	2000	477

Experimental Settings



- Model Settings
  - 1 Convolutional Layer and 1 Pooling Layer
  - 128 hidden nodes for the MLP
  - ReLU as the activation function

- Impact of Similarity Function
- Impact of Kernel Size
- Impact of Dynamic Pooling Size

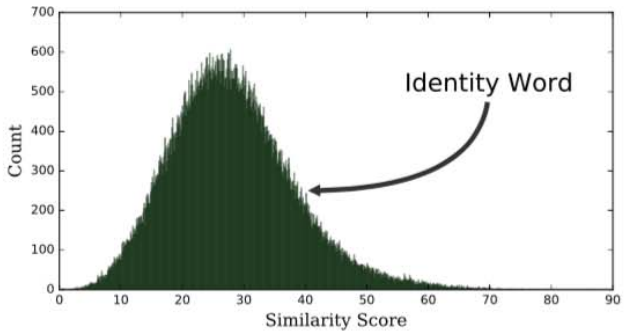
Impact of Similarity Function

Comparison of different similarity functions on Paraphrase Identification

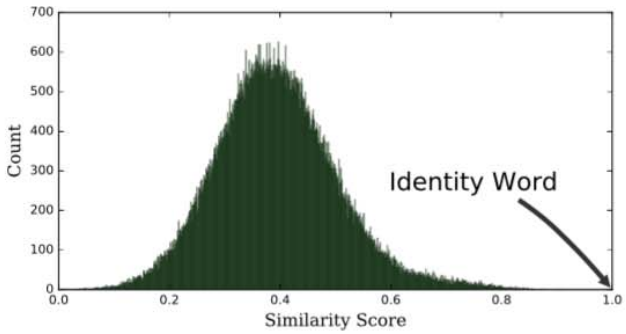
Model	Acc.(%)	F_1(%)
MP-IND	75.77	82.66
MP-COS	75.13	82.45
MP-DOT	**75.94**	**83.01**

Comparison of different similarity functions on Robust04

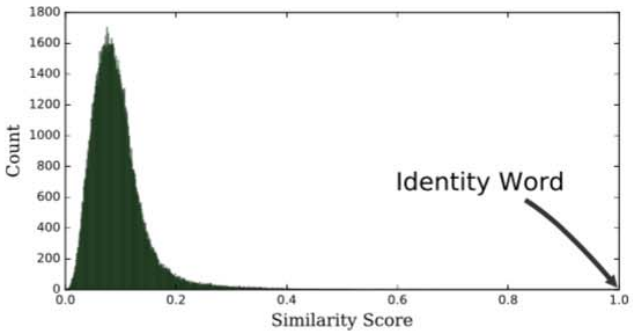
Model	MAP	nDCG@20	P@20
MP-Ind	0.225	0.387	0.302
MP-Dot	0.095	0.149	0.142
MP-Cos	0.189	0.340	0.272
MP-Gau	**0.226**	**0.403**	**0.318**



(a) Dot Product



(b) Cosine



(c) Gaussian Kernel

It is important for deep models to differentiate exact matching from semantic matching on ad-hoc retrieval task

Impact of Kernel Size

- **1 x n kernel**: to capture the query word level richer matching signals
- **n x n kernel**: to capture the n-gram matching (word proximity)

- **Observations**:
  - Kernel size is not sensitive on the Indicator Matching Matrix due to sparsity.
  - A proper kernel size on the Gaussian Matching Matrix can capture richer semantic information.
  - **Surprisingly, no improvement on n-gram matching**: The dataset is too small to learn the complex proximity patterns.

Table 4: Comparison of different kernel size of MatchPyramid. (fix pooling size 3×10)

Model	Kernel Size	MAP	nDCG@20	P@20
MP-Ind	1×1	0.225	**0.387**	**0.302**
MP-Ind	1×3	**0.226**	0.382	0.294
MP-Ind	1×5	0.223	0.382	0.297
MP-Ind	3×3	0.221	0.379	0.295
MP-Ind	5×5	0.219	0.378	0.295
MP-Gau	1×1	0.226	0.403	0.318
MP-Gau	1×3	**0.232**	**0.411**	**0.327**
MP-Gau	1×5	0.226	0.409	0.326
MP-Gau	3×3	0.220	0.400	0.312
MP-Gau	5×5	0.201	0.371	0.301

Impact of Pooling Size

Pooling layer is used to pick out the most important signals

Observation:

1. Pooling size affect the performance significantly
2. The best dynamic pooling size is about 3x10
   - On the query side, we keep information **at query word level** (since the mean query length is 3);
   - On the document side, we aggregate information from every 50 words, close to **the average length of a paragraph**.

Table 2: Comparison of different pooling size of MatchPyramid. (fix kernel size 1×1)

Model	Pooling Size	MAP	nDCG@20	P@20
MP-Ind	5×100	0.175	0.320	0.254
MP-Ind	5×50	0.195	0.343	0.266
MP-Ind	5×20	0.209	0.363	0.279
MP-Ind	5×10	**0.219**	**0.387**	**0.301**
MP-Ind	5×5	0.214	0.380	0.295
MP-Ind	5×3	0.209	0.370	0.300
MP-Ind	3×20	0.213	0.357	0.285
MP-Ind	3×10	**0.225**	**0.387**	**0.302**
MP-Ind	3×5	0.225	0.385	0.301
MP-Ind	3×3	0.215	0.377	0.301
MP-Ind	1×10	0.056	0.082	0.073
MP-Ind	1×5	0.051	0.072	0.078
MP-Ind	1×3	0.043	0.066	0.053

It is better to aggregate signals for the lengthy, noisy documents but not the short queries

Compare with Existing Models

- MatchPyramid outperforms existing deep matching models
  - Compared with representation based models, MatchPyramid keeps all the low-level detailed matching signals
  - Compared with interaction based models, Gaussian kernel is much more effective than 1D convolution in ARC-II
- The best performing deep matching models cannot beat the traditional retrieval models on Robust04

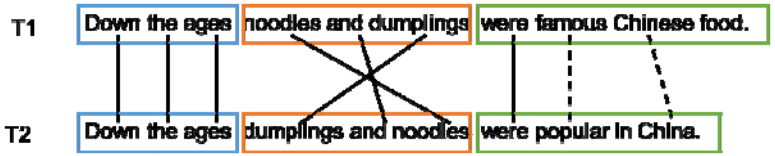
Table 5: Comparison of different retrieval models over the TREC collection Robust04. † models trained on large click-through dataset.

Type	Name	MAP	nDCG@20	P@20
Traditional	QL	0.253	0.415	0.369
Model	BM25	**0.255**	**0.418**	**0.370**
Representation	DSSM†	0.095	0.201	0.171
Based Model	CDSSM†	0.067	0.146	0.125
	ARC-I	0.041	0.066	0.065
Interaction	ARC-II	0.067	0.147	0.128
Based Model	MP-Gau	0.232	0.411	0.327

Some Thoughts…

- Considering the success of these deep matching models on NLP tasks, e.g. paraphrase identification
  - **Task:** It seems that the ad-hoc retrieval task is quire different from those text matching tasks in NLP:

Semantic matching

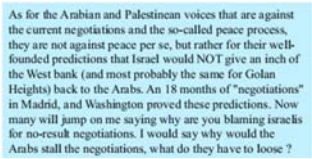


VS.

Relevance matching

Q: peace process

D:

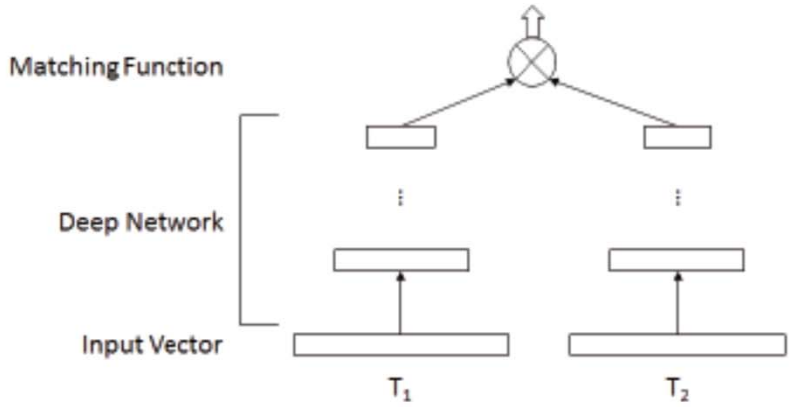

As for the Arabian and Palestinean voices that are against the current negotiations and the so-called peace process, they are not against peace per se, but rather for their well-founded predictions that Israel would NOT give an inch of the West bank (and most probably the same for Golan Heights) back to the Arabs. An 18 months of "negotiations" in Madrid, and Washington proved these predictions. Now many will jump on me saying why are you blaming israelis for no-result negotiations. I would say why would the Arabs stall the negotiations, what do they have to loose ?

  - Homogeneous texts (sentences) vs. heterogeneous texts (keywords, documents)
  - Comparable lengths vs. significantly different lengths (short queries, very long documents)
  - Semantic matching signals vs. exact matching signals

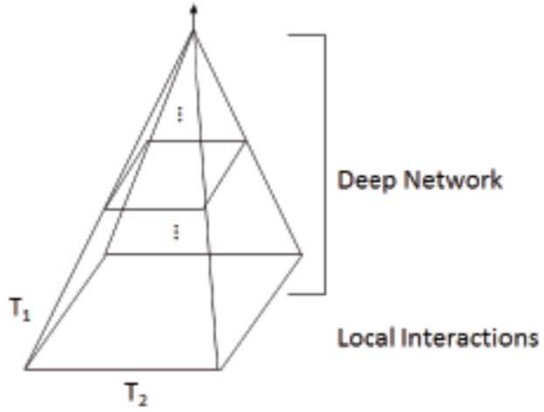
A Deep Relevance Matching Model for Ad-hoc Retrieval, to appear in CIKM 2016

Some Thoughts…

- Considering the success of traditional retrieval models,
  - **Model:** Queries are usually not treated equivalent to documents in traditional retrieval models
    - VSM (symmetric) vs. BM25/LM (non-symmetric)
    - Most deep models treat the two input texts in a symmetric way



DSSM/CDSSM/ARC I



ARC II/MatchPyramid

Some Thoughts…

- Considering the success of deep learning models in other research areas,
  - **Data:** The limited data size (i.e., query size) of ad-hoc retrieval dataset prevents us from using very deep network architectures

Table 1: Statistics of the TREC collection Robust04.

#Vocab	#Doc	#Query	#Retrieval doc	Avg doc
0.6M	0.5M	250	2000	477

  - Some larger data sets:
    - MQ2007: about 1700 queries labeled
    - MQ2008: about 800 queries labeled
  - Paraphrase Identification: 5,800 pairs of sentences (MSRP)
  - ImageNet: 14,197,122 images, synsets: 21,841

Conclusions

- Test MatchPyramid models on ad-hoc retrieval tasks
- Study three factors in MatchPyramid models
  - Similarity functions
  - Kernel size
  - Dynamic pooling size
- The weak performance on ad-hoc retrieval tasks
  - Task characteristics
  - Model design
  - Data size

Thanks

Q&A