ТЕХНОСФЕРА

Learning to match 2

Владимир Гулин

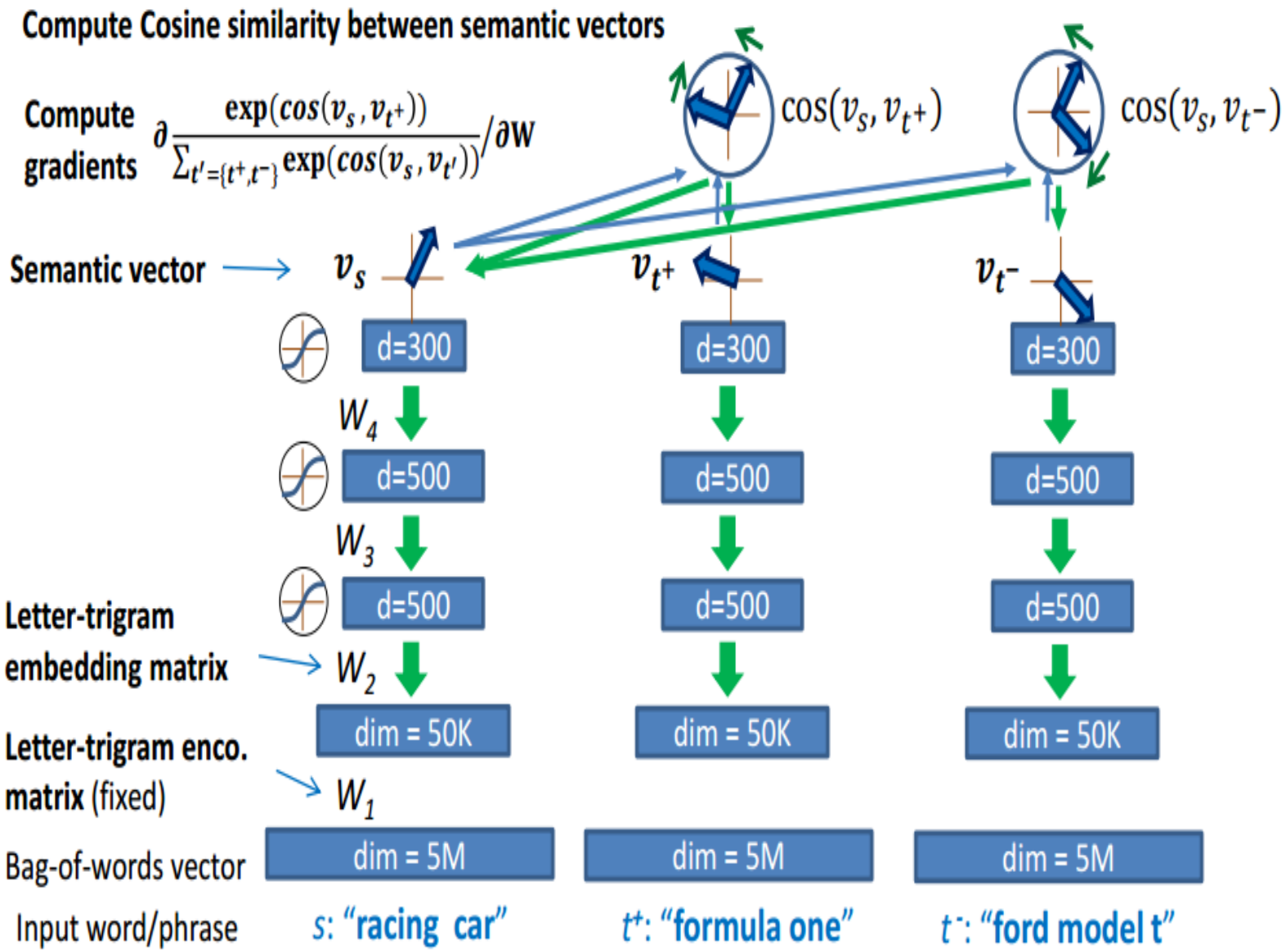
27 ноября 2019 г.

План лекции

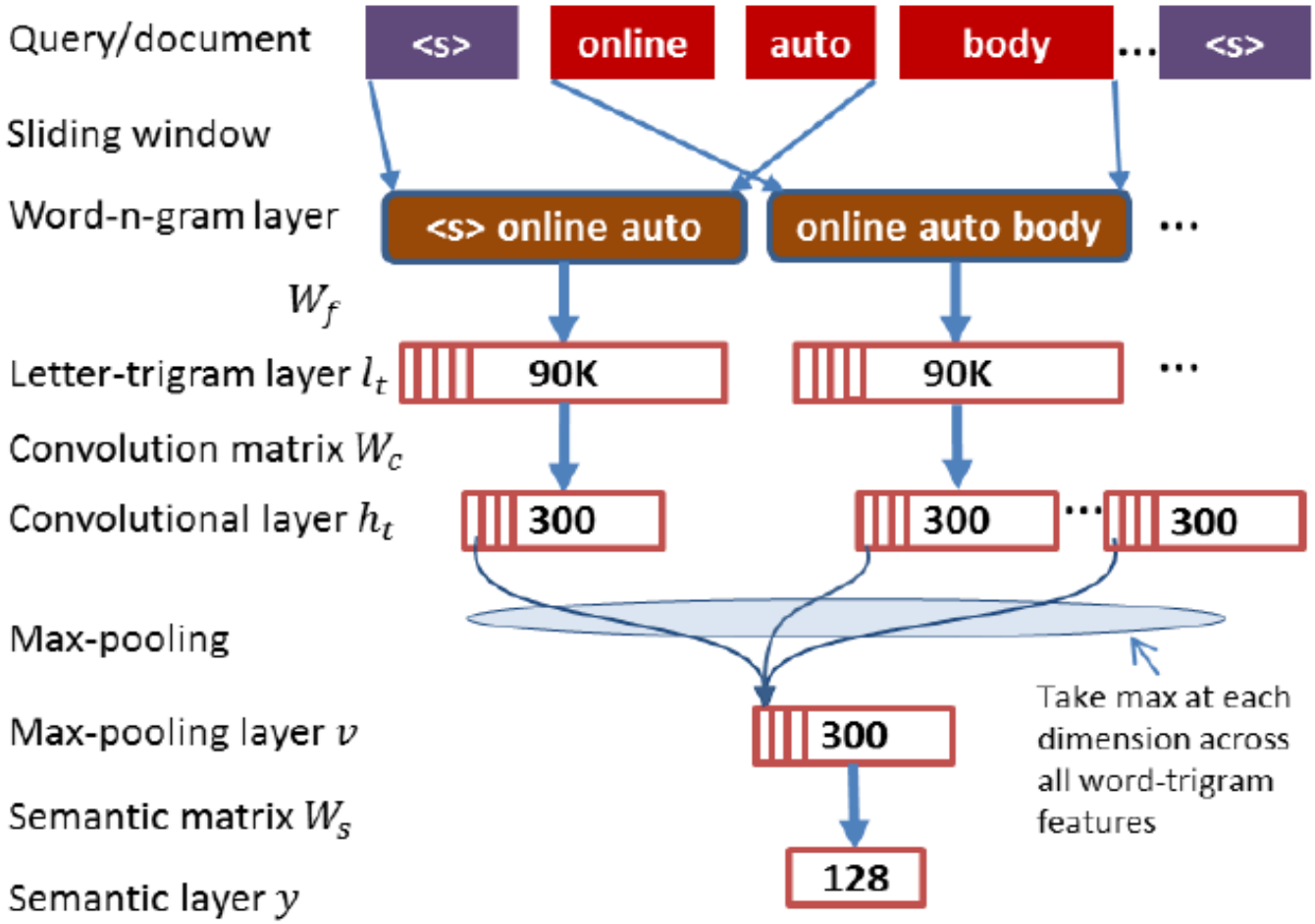
Напоминание

Обзор современных статей

DSSM


Compute Cosine similarity between semantic vectors
Compute gradients
$\partial \frac{\exp(cos(v_s, v_{t^+}))}{\sum_{t'=\{t^+,t^-\}} \exp(cos(v_s, v_{t'}))} / \partial W$
$\cos(v_s, v_{t^+})$
$\cos(v_s, v_{t^-})$
Semantic vector
v_s
v_{t^+}
v_{t^-}
d=300
d=300
d=300
W_4
d=500
d=500
d=500
W_3
d=500
d=500
d=500
Letter-trigram embedding matrix
W_2
dim = 50K
dim = 50K
dim = 50K
Letter-trigram enco. matrix (fixed)
W_1
Bag-of-words vector
dim = 5M
dim = 5M
dim = 5M
Input word/phrase
s: "racing car"
t^+: "formula one"
t^-: "ford model t"

CLSM



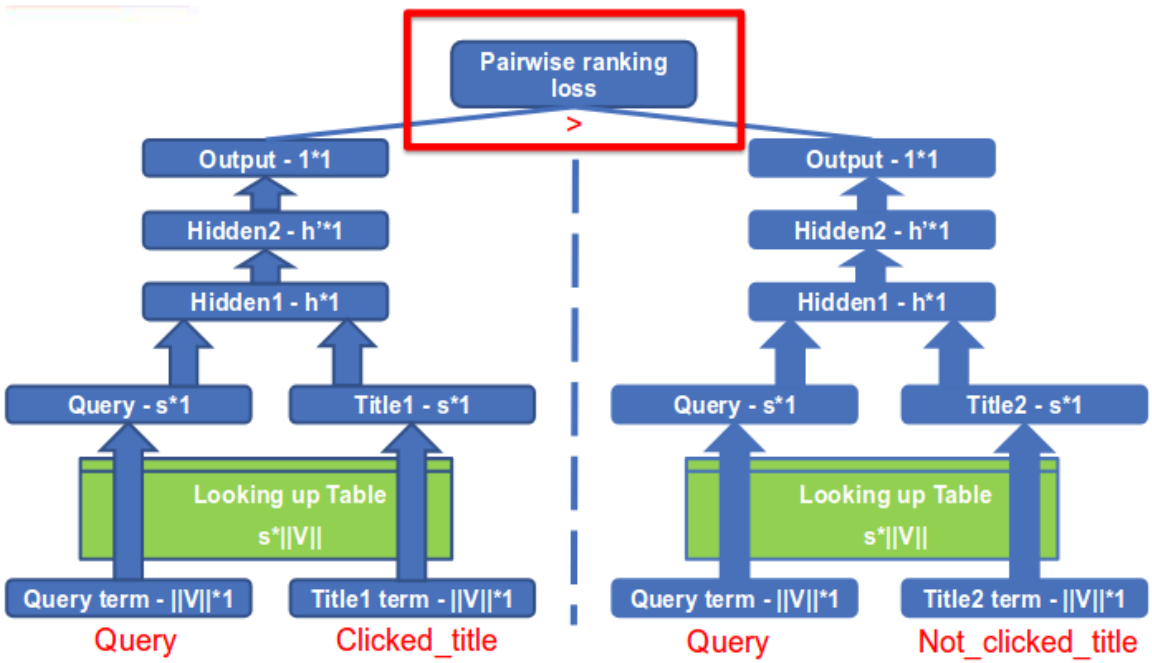
Вопрос:

- Как такие модели обучать?

Industry

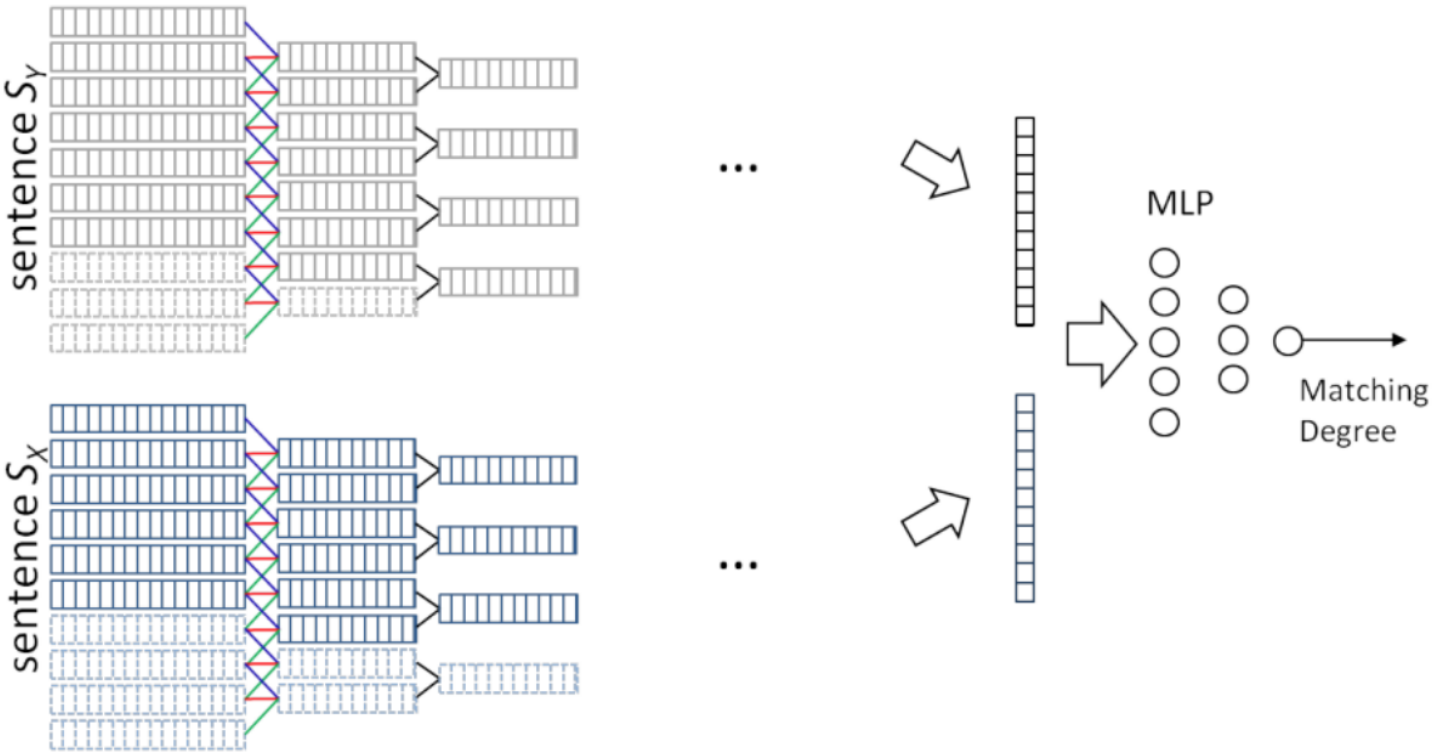
Baidu DNN model

- Порядка 30% в улучшении качества за 2013 и 2014
- Используют 10 млрд. кликов для обычения (более 100 млн. параметров)



ARC-I

- Similar to DSSM, perform 1D convolution on text representations separately.
- Originally introduced for paraphrasing task.

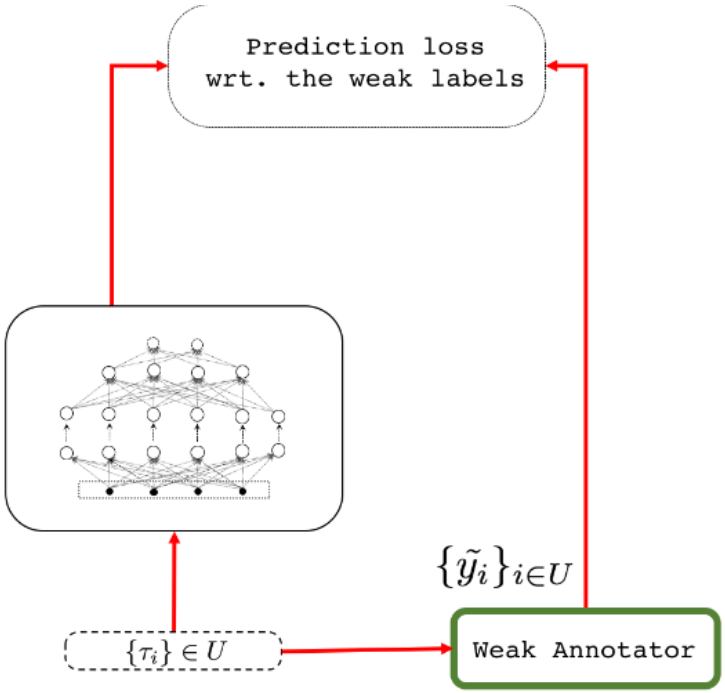


Training neural networks using pseudo relevance

Main idea

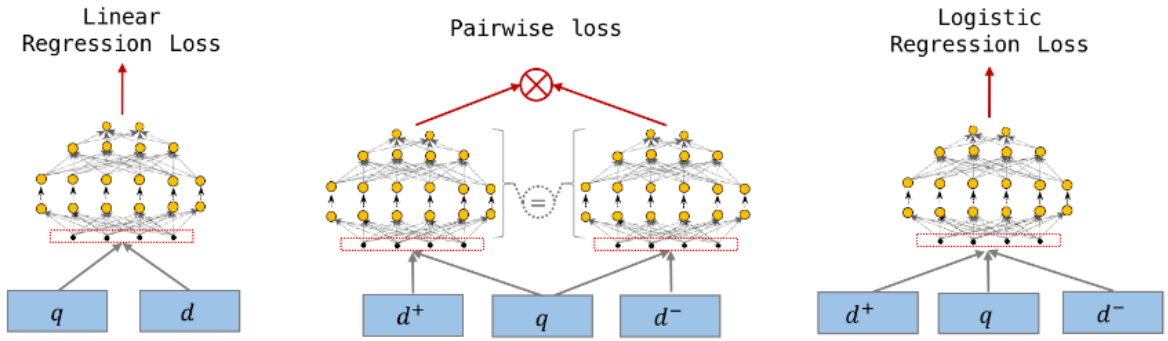
Размечаем большое количество неразмеченных данных использую слабую модель (Pseudo-Labeling) и тренируем модель на таком слабом таргете

- Аппроксимируем BM25
- Побеждаем BM25 обучившись на BM25!



Training neural networks using pseudo relevance

- Используются 3 различные архитектуры Score, Rank, RankProb



- Моделям скармливаются три различных представления

- Dense: $\psi(q,d) = [N||avg(l_d)_D||l_d||\{df(t_i)||tf(t_i,d)\}_{1\leq i\leq k}]$
- Sparse: $\psi(q,d) = [tfv_c||tfv_q||tfv_d]$
- Embedding-based: $\psi(q,d) = [\odot_{i=1}^{|q|}(\mathcal{E}(t_i^q),\mathcal{W}(t_i^q))||\odot_{i=1}^{|d|}(\mathcal{E}(t_i^d),\mathcal{W}(t_i^d))]$

Training neural networks using pseudo relevance

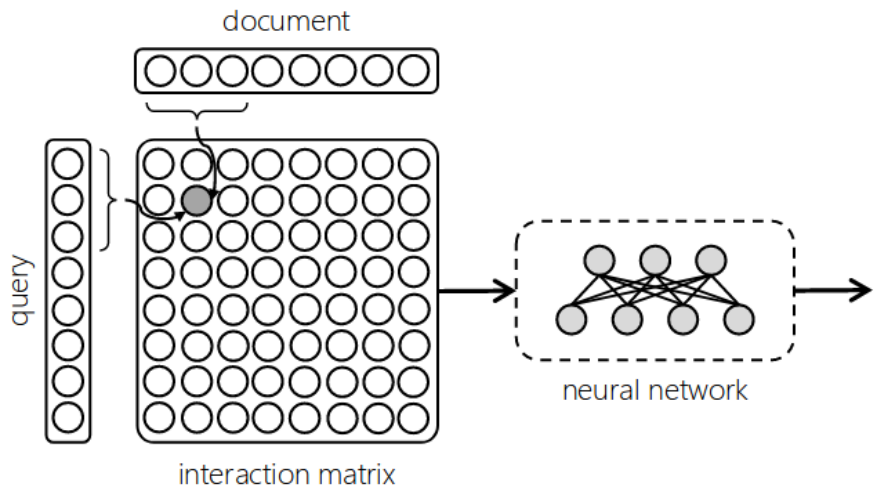
Method	Robust04			ClueWeb		
	MAP	*P@20*	*nDCG@20*	*MAP*	*P@20*	*nDCG@20*
BM25	0.2503	0.3569	0.4102	0.1021	0.2418	0.2070
Score + Dense	0.1961▽	0.2787▽	0.3260▽	0.0689▽	0.1518▽	0.1430▽
Score + Sparse	0.2141▽	0.3180▽	0.3604▽	0.0701▽	0.1889▽	0.1495▽
Score + Embed	0.2423▽	0.3501	0.3999	0.1002	0.2513	0.2130
Rank + Dense	0.1940▽	0.2830▽	0.3317▽	0.0622▽	0.1516▽	0.1383▽
Rank + Sparse	0.2213▽	0.3216▽	0.3628▽	0.0776▽	0.1989▽	0.1816▽
Rank + Embed	**0.2811**▲	**0.3773**▲	**0.4302**▲	**0.1306**▲	**0.2839**▲	**0.2216**▲
RankProb + Dense	0.2192▽	0.2966▽	0.3278▽	0.0702▽	0.1711▽	0.1506▽
RankProb + Sparse	0.2246▽	0.3250▽	0.3763▽	0.0894▽	0.2109▽	0.1916
RankProb + Embed	**0.2837**▲	**0.3802**▲	**0.4389**▲	**0.1387**▲	**0.2967**▲	**0.2330**▲

Interaction matrix based approaches

Alternative to Siamese networks

Interaction matrix X, where $x_{i,j}$ is obtained by comparing the the i^{th} word in source sentence with j^{th} word in target sentence

Comparisons can be both lexical or semantic

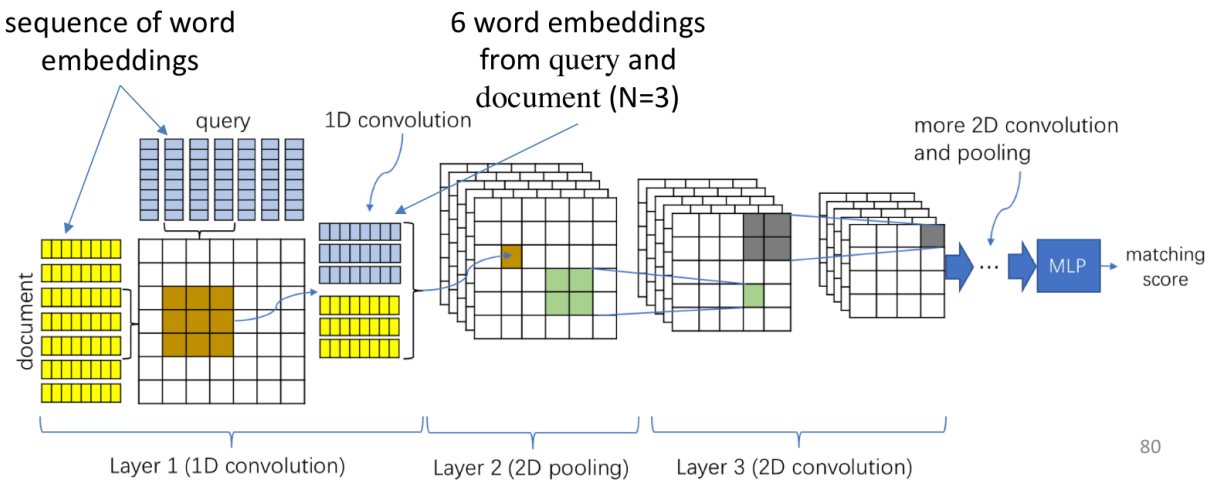


Вопрос:

- Как такое использовать с линейными моделями?

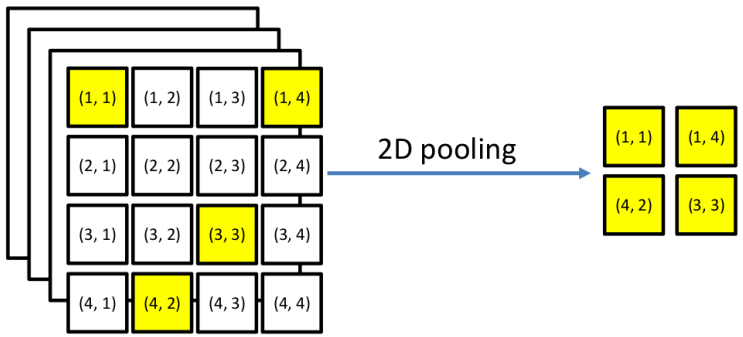
ARC-II

- Let two sentences meet before their own high-level representations mature
- Basic matching signals: phrase sum interaction matrix
- Interaction: CNN to capture the local interaction structure
- Aggregation Function: MLP



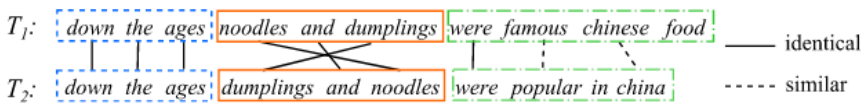
ARC-II

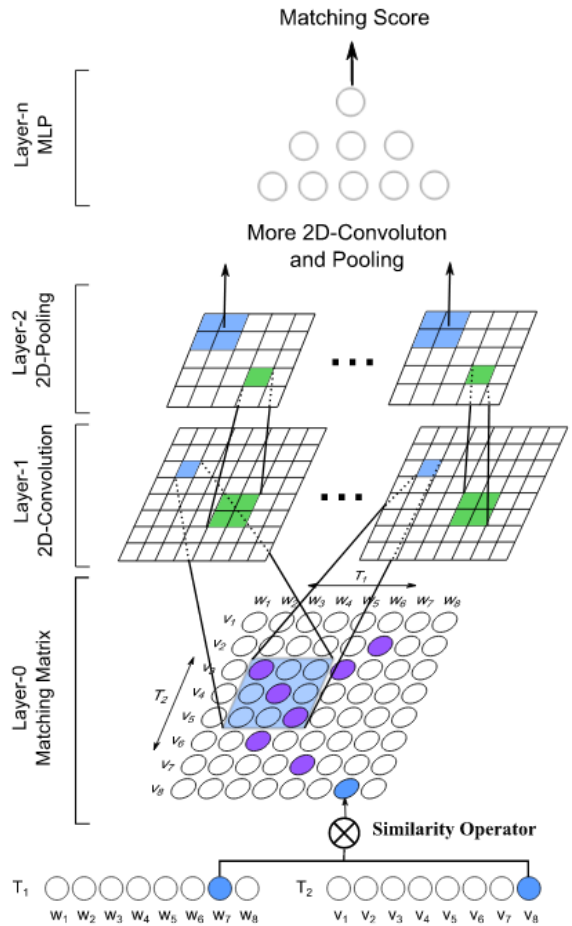
- Keeping word order information
  - Both the convolution and pooling are order preserving



- However, word level exact matching signals are lost
  - 2-D matching matrix is constructed based on the embedding of the words in two N-grams

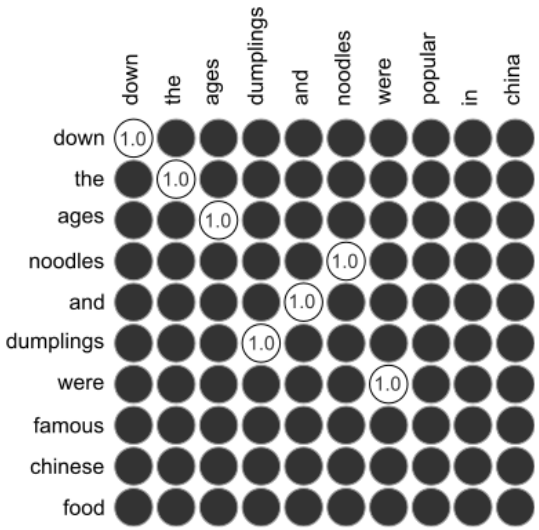
MatchPyramid



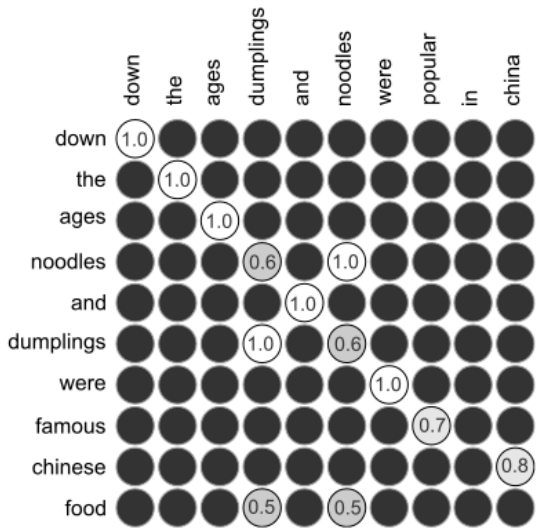


MatchPyramid

Interaction matrix for two texts captures pairwise similarity between their words

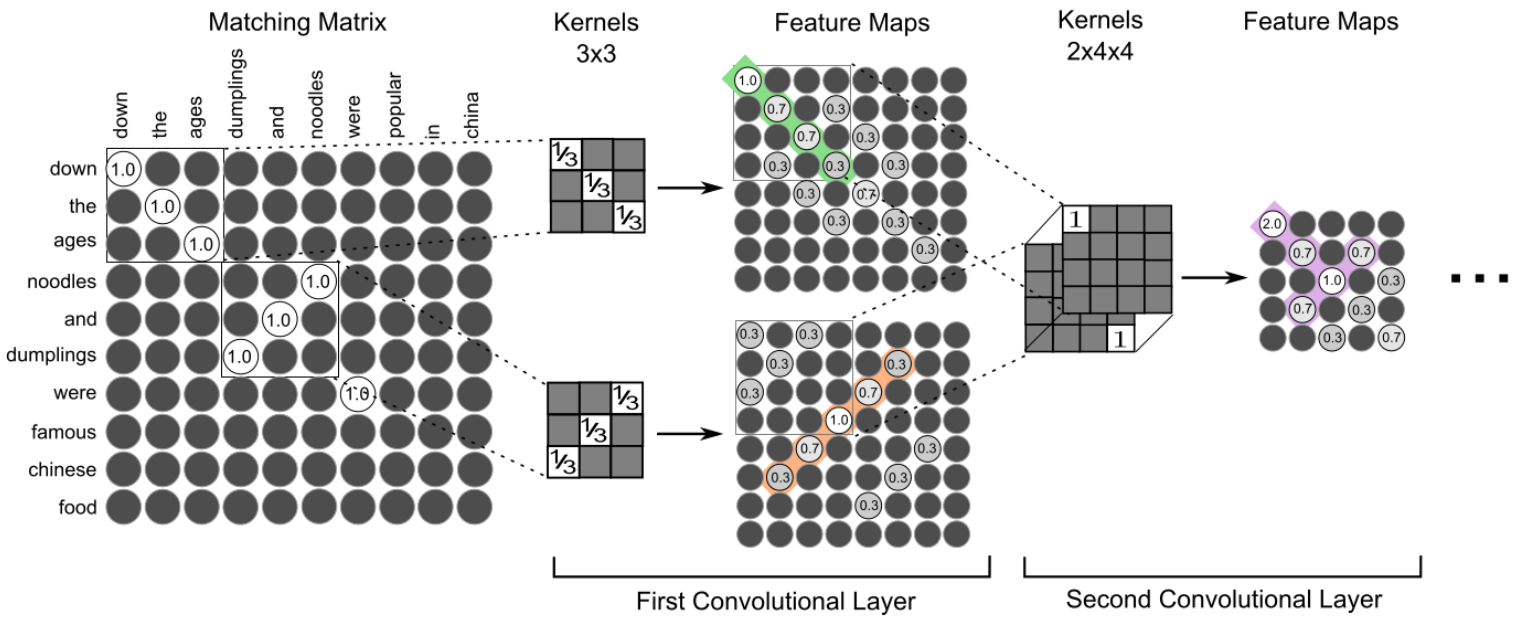


Exact match

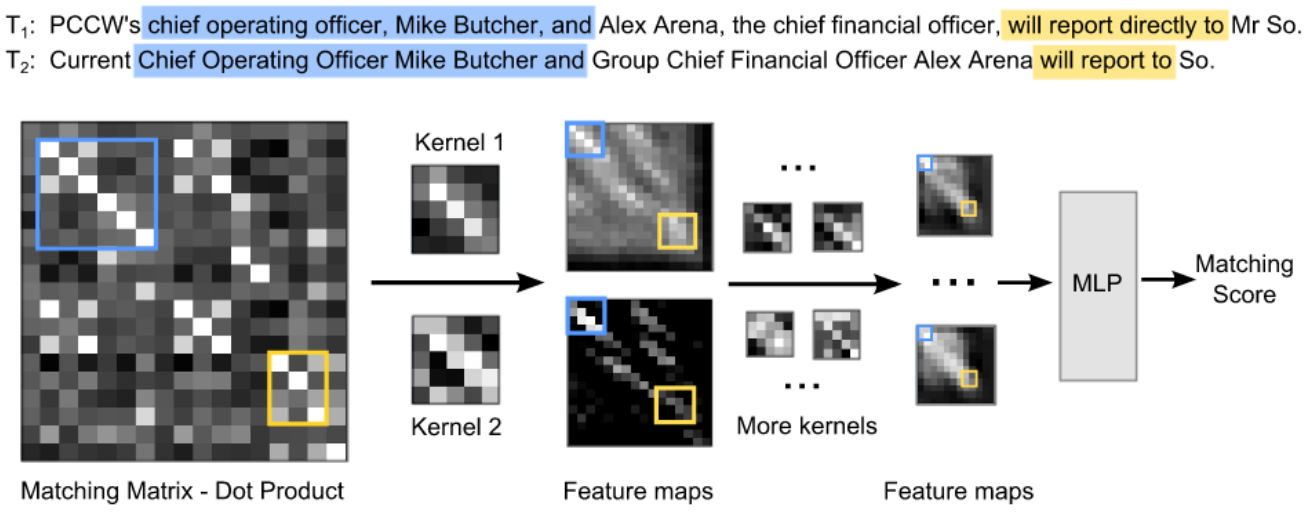


Cosine similarity

Interaction Matrix: MatchPyramid



MatchPyramid



MatchPyramid

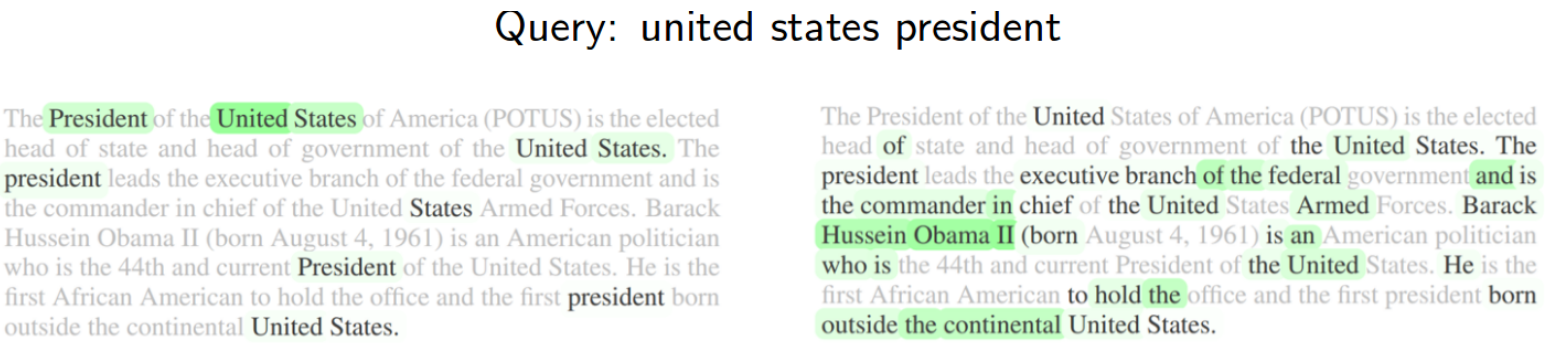
Table 1: Results on MSRP.

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**

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**

Lexical and Semantic matching



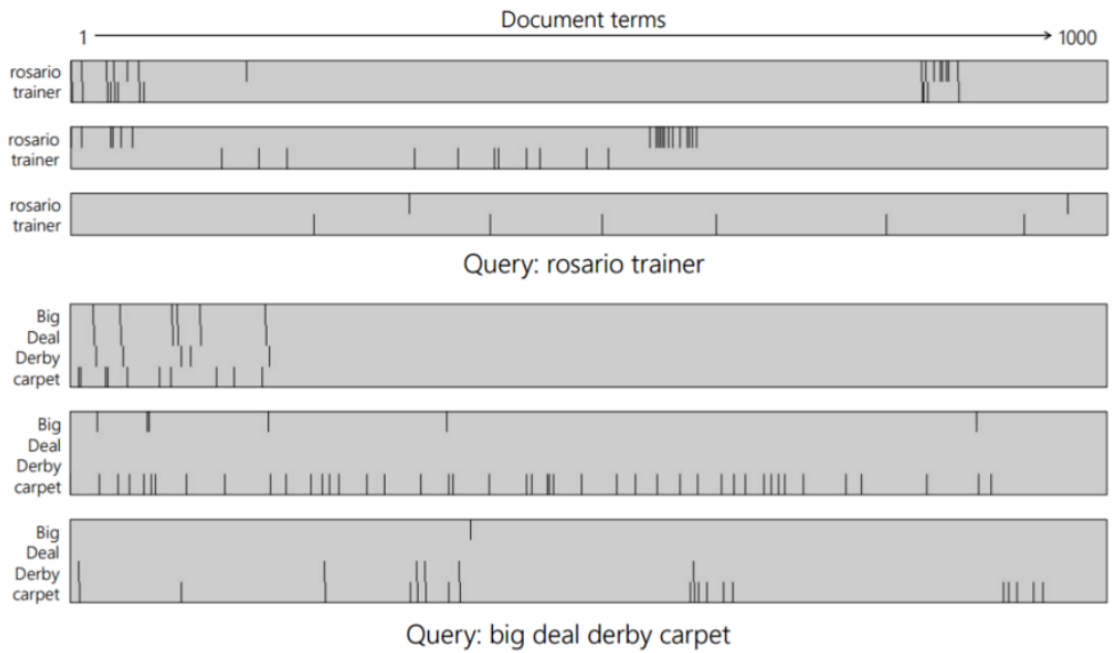
- Традиционные модели инфопоиска работают исключительно на лексическом матчинге
- Representation модели могут учитывать как все термы документа соотносятся с запрсом
- И те и другие могут быть смоделированы с помощью нейросетей

Lexical Matching

Query: "rosario trainer"

The rare term "rosario" may have never been seen during training and unlikely to have meaningful representation

But the patterns of lexical matches of rare terms in document may be very informative for estimating relevance

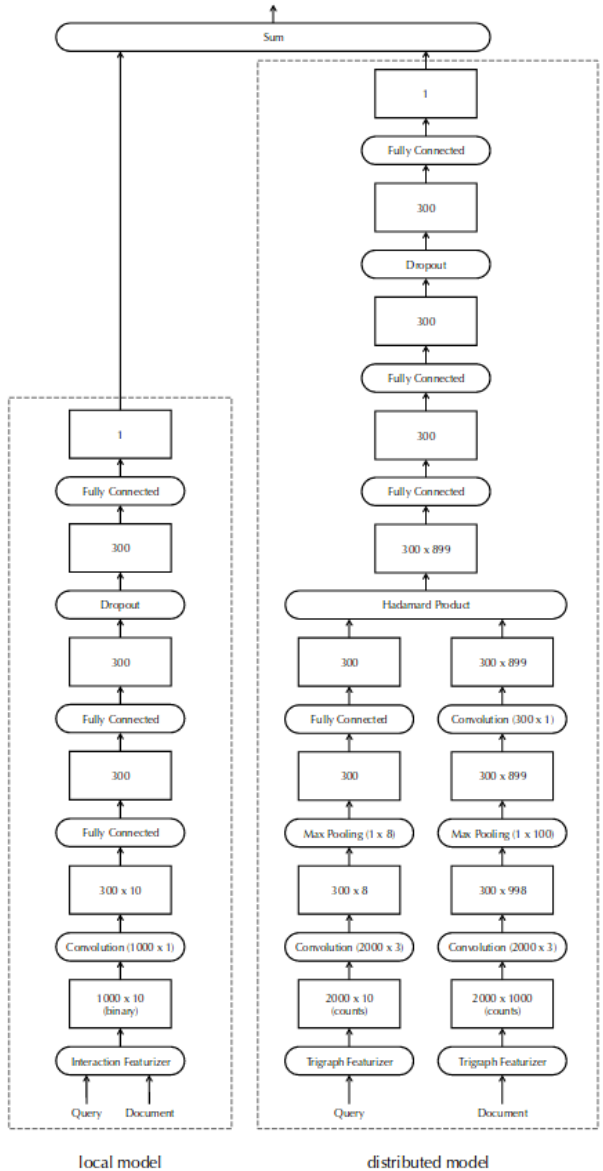


Duet Model

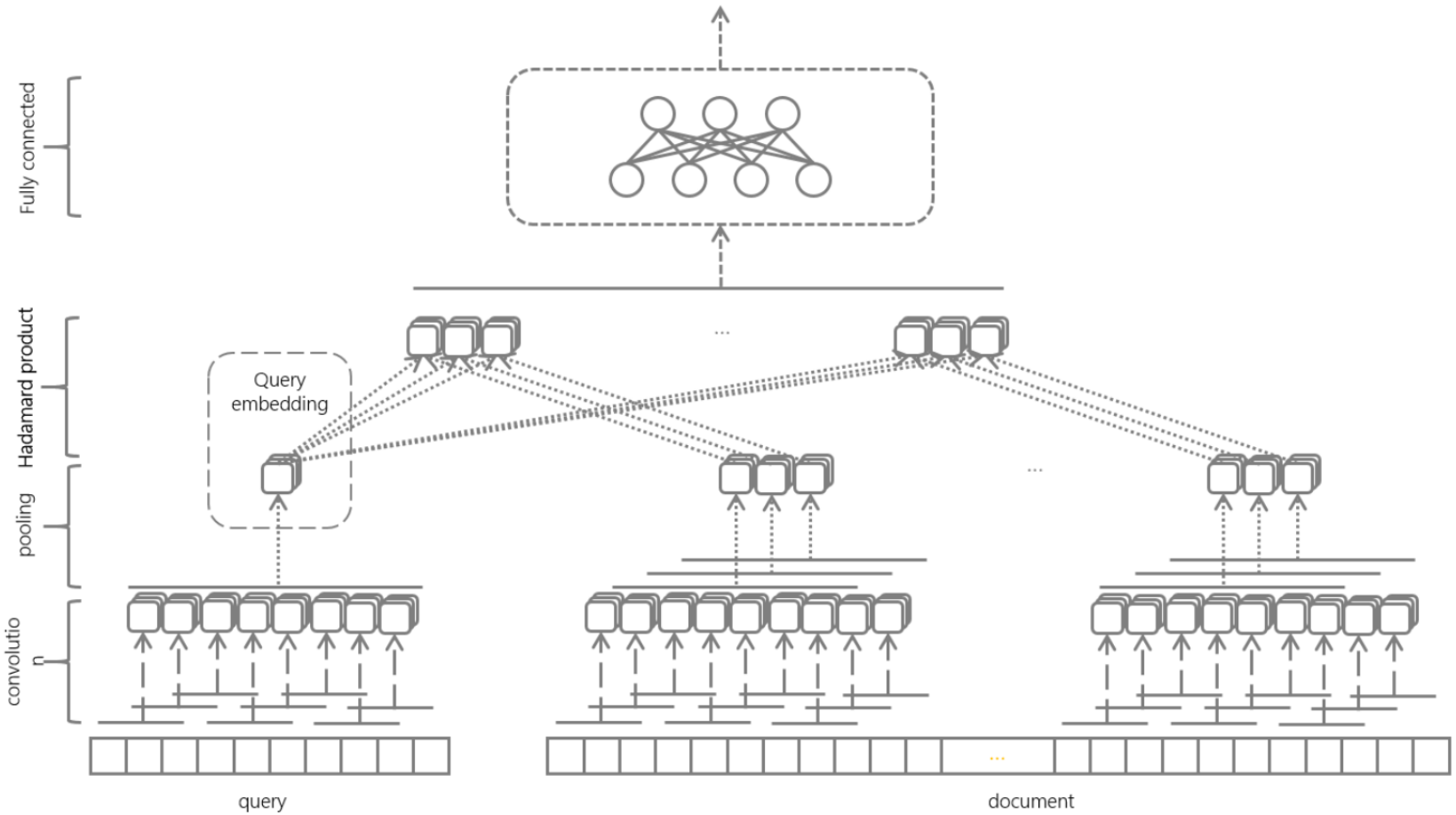
Совместно тренируем 2 подсетки, сфокусированных на лексическом и семантическом матчинге

Training sample:
$q, d^+, d_1, d_2, d_3, d_4$

$$p(d^+|q) = \frac{e^{nrdm(q,d^+)}}{\sum_{d \in D^-} e^{nrdm(q,d)}}$$



Distributed Model



Duet Model

(a) weighted

	NDCG@1	NDCG@10
Non-neural baselines		
LSA	22.4	44.2
BM25	24.2	45.5
DM	24.7	46.2
QL	24.6	46.3
Neural baselines		
DRMM	24.3	45.2
DSSM	25.8	48.2
CDSSM	*27.3*	48.2
DESM	25.4	*48.3*
Our models		
Local model	24.6	45.1
Distributed model	28.6	50.5
Duet model	**32.2**	**53.0**

(b) unweighted

	NDCG@1	NDCG@10
Non-neural baselines		
LSA	31.9	62.7
BM25	34.9	63.3
DM	35.0	63.4
QL	34.9	63.4
Neural baselines		
DRMM	*35.6*	*65.1*
DSSM	34.3	64.4
CDSSM	34.3	64.0
DESM	35.0	64.7
Our models		
Local model	35.0	64.4
Distributed model	35.2	64.9
Duet model	**37.8**	**66.4**

Term importance

Local model

Only query terms have an impact

Earlier occurrences have bigger impact

Query: united states president

The President of the United States of America (POTUS) is the elected head of state and head of government of the United States. The president leads the executive branch of the federal government and is the commander in chief of the United States Armed Forces. Barack Hussein Obama II (born August 4, 1961) is an American politician who is the 44th and current President of the United States. He is the first African American to hold the office and the first president born outside the continental United States.

Visualizing impact of dropping terms on model score

Duet Model

Term importance

Distributed model

Non-query terms (e.g., *Obama* and *federal*) has positive impact on score

Common words like 'the' and 'of' probably good indicators of well-formedness of content

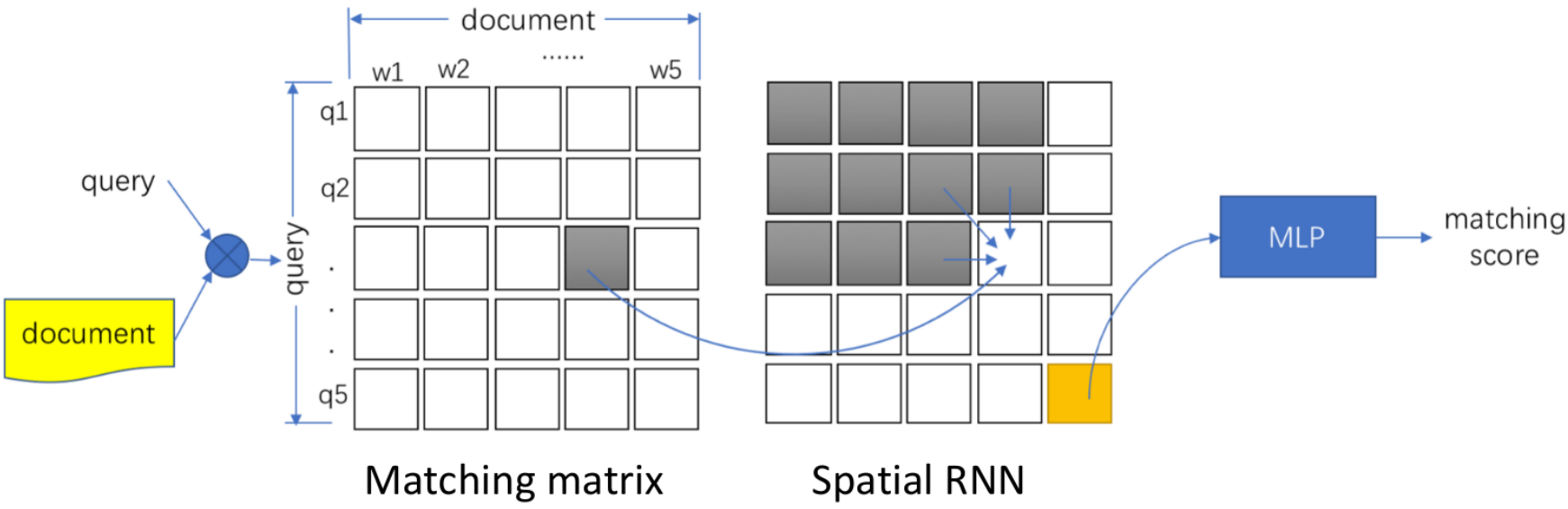
Query: united states president

The President of the United States of America (POTUS) is the elected head of state and head of government of the United States. The president leads the executive branch of the federal government and is the commander in chief of the United States Armed Forces. Barack Hussein Obama II (born August 4, 1961) is an American politician who is the 44th and current President of the United States. He is the first African American to hold the office and the first president born outside the continental United States.

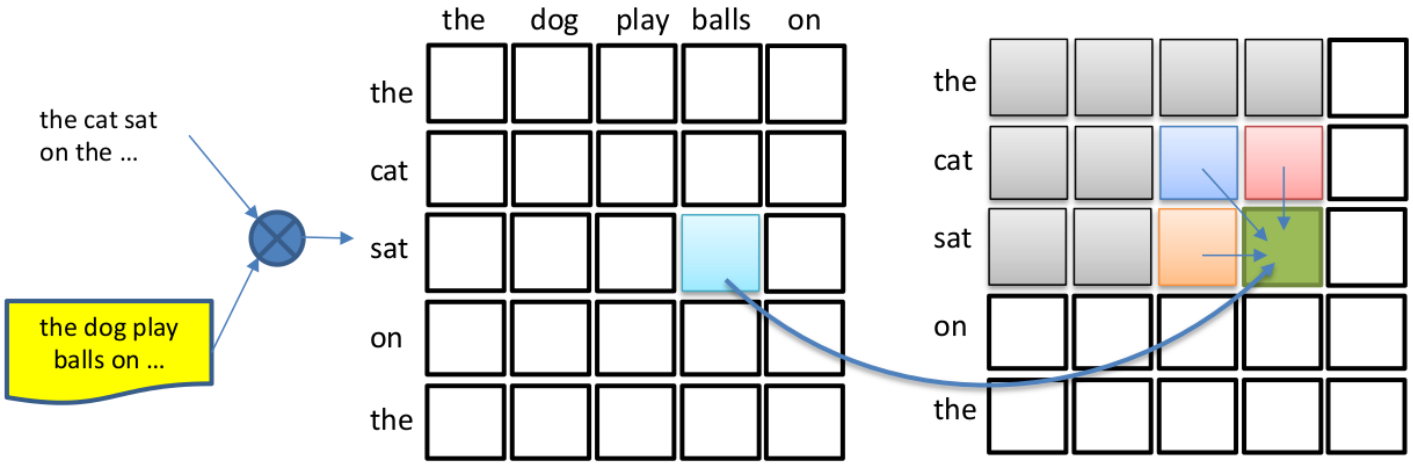
Visualizing impact of dropping terms on model score

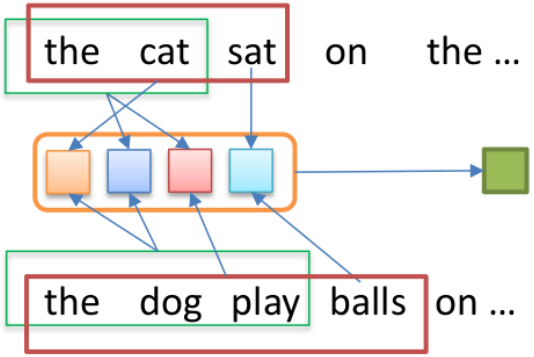
Match-SRNN

- Based on spatial recurrent neural network (SRNN)
- Basic matching signals: word-level matching matrix
- Matching function: Spatial RNN + MLP



Match-SRNN



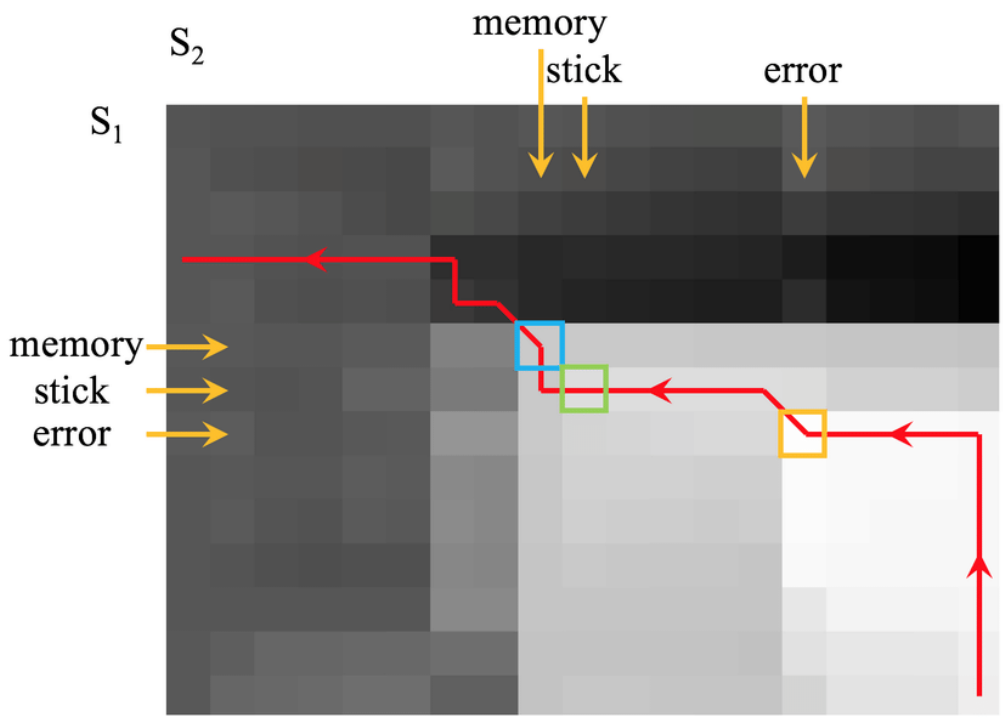


- Calculated recursively (from top left to bottom right)
- All matching signals between the prefixes been utilized
  - **Current position**: sat <—> balls
  - **Substrings**:
    - the cat <—> the dog play
    - the cat <—> the dog play balls
    - the cat sat <—> the dog play

Match-SRNN

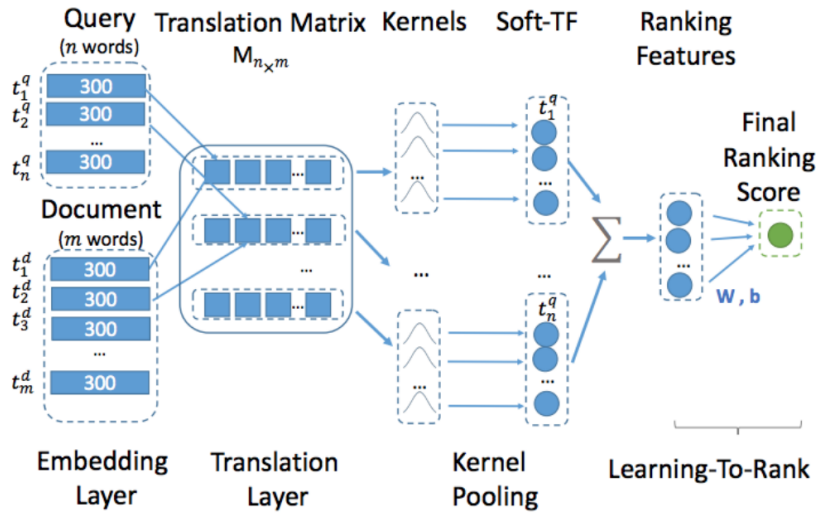
*Question: "How to get rid of **memory stick error** of my sony cyber shot?"*

*Answer: "You might want to try to format the **memory stick** but what is the **error** message you are receiving."*

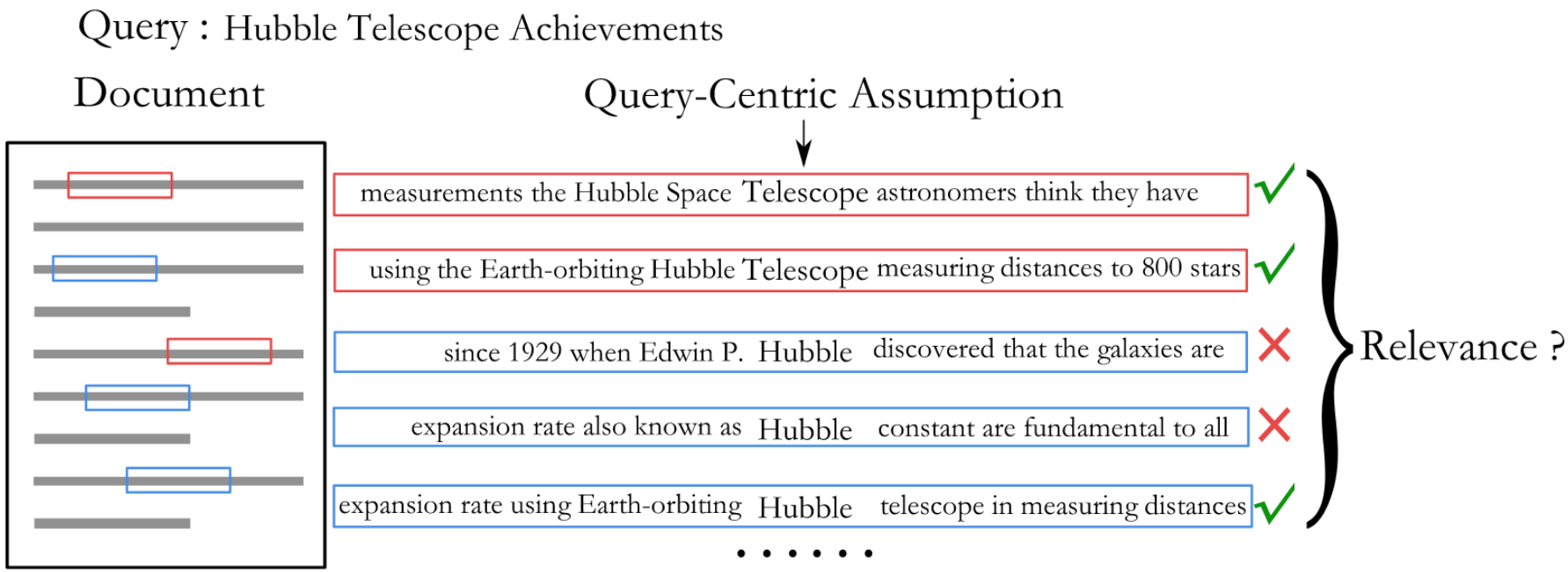


K-NRM

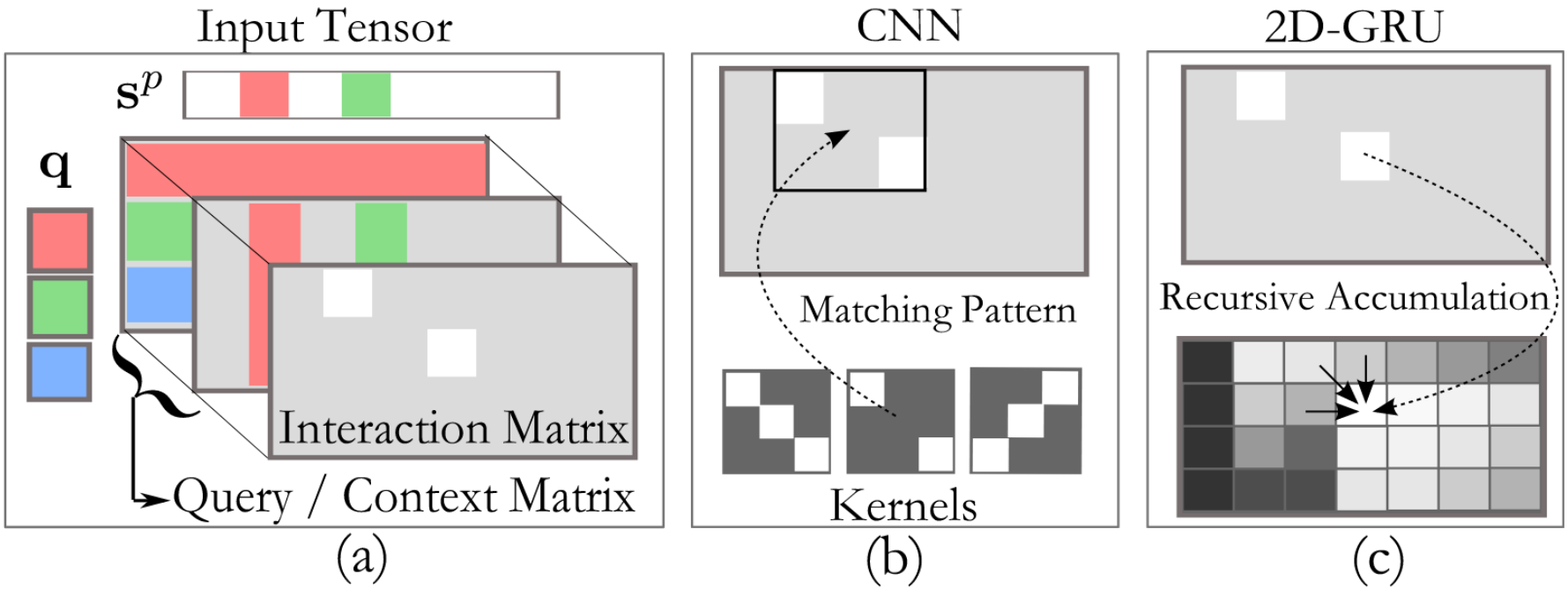
- Compute word-interaction matrix, apply k kernels to every query term row in interaction matrix.
- This results in k-dimensional vector.
- Aggregate the query term vectors into a fixed-dimensional query representation.



DeepRank

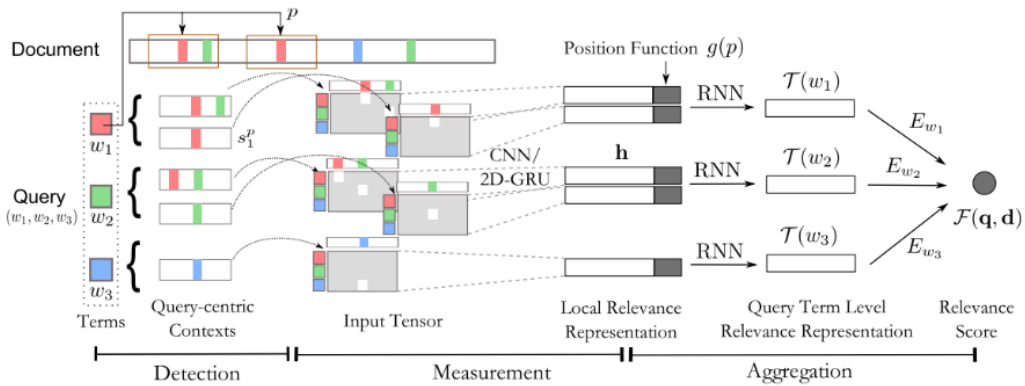

Query : Hubble Telescope Achievements
Document
Query-Centric Assumption
measurements the Hubble Space Telescope astronomers think they have
using the Earth-orbiting Hubble Telescope measuring distances to 800 stars
since 1929 when Edwin P. Hubble discovered that the galaxies are
expansion rate also known as Hubble constant are fundamental to all
expansion rate using Earth-orbiting Hubble telescope in measuring distances
Relevance ?

DeepRank


Input Tensor
$\mathbf{s}^p$
$\mathbf{q}$
Interaction Matrix
Query / Context Matrix
(a)
CNN
Matching Pattern
Kernels
(b)
2D-GRU
Recursive Accumulation
(c)

DeepRank

- Focus only on exact term occurrences in document.
- Compute interaction between query and window surrounding term occurrence.
- RNN or CNN then combines per-window features (query representation, context representations and interaction between query/document term) into matching score.



DeepRank

	MQ2008								
Model	NDCG@1	NDCG@3	NDCG@5	NDCG@10	P@1	P@3	P@5	P@10	MAP
BM25-Title	0.344^{-}	0.420^{-}	0.461^{-}	0.220^{-}	0.408^{-}	0.381^{-}	0.337^{-}	0.245^{-}	0.465^{-}
RankSVM	0.375^{-}	0.431^{-}	0.479^{-}	0.229	0.441^{-}	0.390^{-}	0.348^{-}	0.249	0.478^{-}
RankBoost	0.381	0.436^{-}	0.477^{-}	0.231	0.455	0.392^{-}	0.347^{-}	0.248	0.481^{-}
AdaRank	0.360^{-}	0.422^{-}	0.462^{-}	0.222	0.430^{-}	0.384^{-}	0.339^{-}	0.247^{-}	0.468^{-}
LambdaMart	0.378	0.437^{-}	0.477^{-}	0.231	0.446^{-}	0.398	0.348^{-}	0.251	0.478^{-}
DSSM	0.286^{-}	0.336^{-}	0.378^{-}	0.178^{-}	0.341^{-}	0.307^{-}	0.284^{-}	0.221^{-}	0.391^{-}
CDSSM	0.283^{-}	0.331^{-}	0.376^{-}	0.175^{-}	0.335^{-}	0.302^{-}	0.279^{-}	0.222^{-}	0.395^{-}
Arc-I	0.295^{-}	0.363^{-}	0.413^{-}	0.187^{-}	0.361^{-}	0.336^{-}	0.311^{-}	0.229^{-}	0.424^{-}
SQA-noFeat	0.291^{-}	0.350^{-}	0.401^{-}	0.184^{-}	0.366^{-}	0.332^{-}	0.309^{-}	0.231^{-}	0.416^{-}
DRMM	0.368^{-}	0.427^{-}	0.468^{-}	0.220^{-}	0.437^{-}	0.392^{-}	0.344^{-}	0.245^{-}	0.473^{-}
Arc-II	0.299^{-}	0.340^{-}	0.394^{-}	0.181^{-}	0.366^{-}	0.326^{-}	0.305^{-}	0.229^{-}	0.413^{-}
MatchPyramid	0.351^{-}	0.401^{-}	0.442^{-}	0.211^{-}	0.408^{-}	0.365^{-}	0.329^{-}	0.239^{-}	0.449^{-}
Match-SRNN	0.369^{-}	0.426^{-}	0.465^{-}	0.223^{-}	0.432^{-}	0.383^{-}	0.335^{-}	0.239^{-}	0.466^{-}
DeepRank-2DGRU	0.391	0.436	0.480	0.236	0.462	0.395	0.354	0.252	0.489
DeepRank-CNN	**0.406**	**0.460**	**0.496**	**0.240**	**0.482**	**0.412**	**0.359**	**0.252**	**0.498**
SQA	0.402	0.454	0.493	0.236	0.485	0.411	0.362	0.254	0.496
DeepRank-CNN-Feat	0.418	0.475	0.507	0.248	0.497	0.422	0.366	0.255	0.508

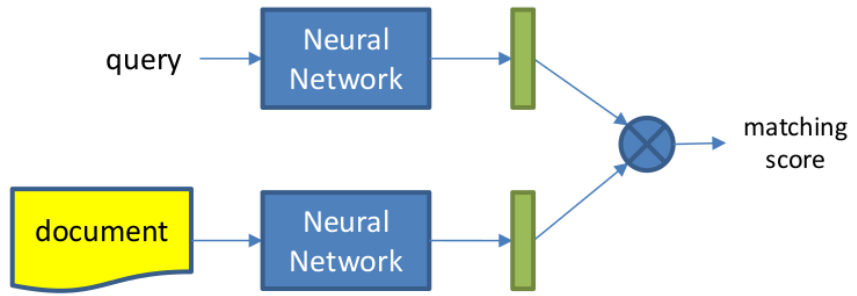
Связь с классикой

$$\mathrm{BM25}(\mathbf{q}, \mathbf{d}) = \sum_{w \in \mathbf{q}} \mathrm{IDF}(w) \cdot \frac{f(w, \mathbf{d}) \cdot (k_1 + 1)}{f(w, \mathbf{d}) + k_1 \cdot (1 - b + \frac{b\,|\mathbf{d}|}{\mathrm{avgdl}})}$$

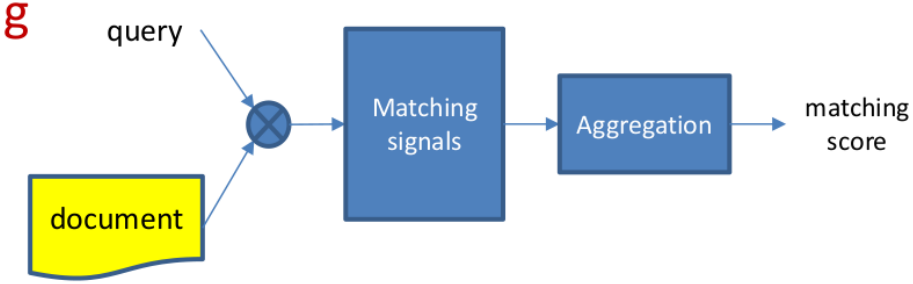
$$\begin{aligned} \mathrm{DeepRank}(\mathbf{q}, \mathbf{d}) &= \sum_{w \in \mathbf{q}} \mathrm{IDF}(w) \cdot \underset{p \in \mathbb{P}(w)}{\mathrm{RNN}} \Big[1, 1/|\mathbf{d}| \Big] \\ &= \sum_{w \in \mathbf{q}} \mathrm{IDF}(w) \cdot \mathcal{G}(f(w, \mathbf{d}), |\mathbf{d}|), \end{aligned}$$

Summary

- Representation learning: representing queries and document in semantic space



- Matching function learning: discovering and aggregating the query-document matching patterns



Вопросы