

# Listwise Neural Ranking Models

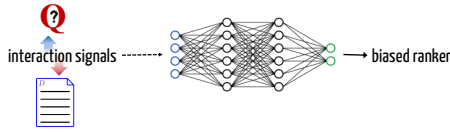
Razieh Rahimi, Ali MontazerAlghaem, James Allan

UMassAmherst

College of Information & Computer Sciences  
Center for Intelligent Information Retrieval

## Problem

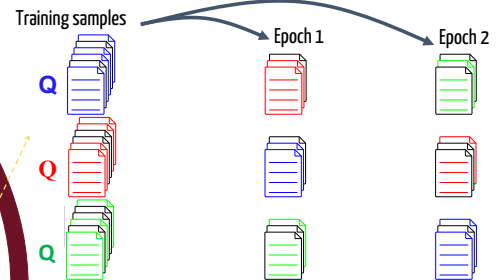
neural ranking models with pointwise or pairwise loss functions that operate on query-document interaction signals can be biased towards queries with more judged or retrieved documents.



## Solution

- Problem: With the same amount of labeled data, learning-to-rank algorithms with pointwise or pairwise loss functions have more training instances than algorithms with listwise loss functions
- Proposed solution: random sampling of documents retrieved for each query before a new epoch of training reshuffling before a new epoch of training

### Query-based sampling and shuffling



## Research Question

Loss functions for learning-to-rank models:

- Pointwise
- Pairwise
- Groupwise
- Listwise

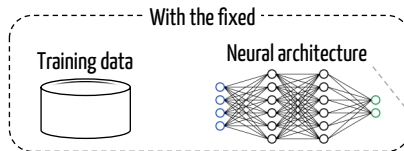
Conventional Learning to Rank Models:

- Operate on human-engineered features
- Better performance using listwise loss functions

RQ: How deep neural ranker with listwise loss function would perform?

## Conclusion

Training a neural ranker with pairwise loss **outperform** Training a neural ranker with listwise loss & query-based sampling



## Network Architecture

Deep Relevance Matching Model (DRMM) → Trained originally with pairwise hinge loss function

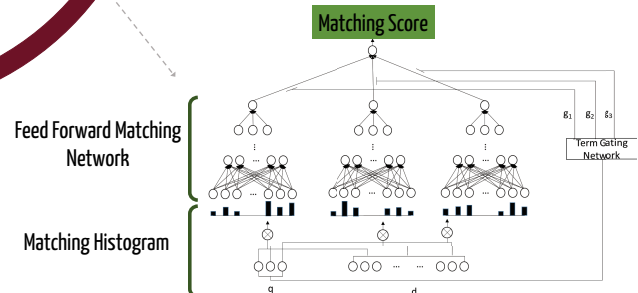
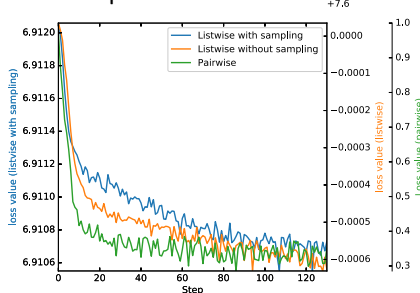


Figure credit: A Deep Relevance Matching Model for Ad-hoc Retrieval, Guo et al., CIKM'16

## Evidence for the conclusion

### Comparison of Loss Functions



## Performance Results

Method	Robust04					ClueWeb				
	P@1	P@3	P@5	P@10	MAP	P@1	P@3	P@5	P@10	MAP
Pairwise loss → DRMM <sub>pl</sub>	0.52	0.4844	0.4613	0.424	0.2923	0.3833	0.3278	0.3367	0.3267	0.2634
Listwise loss → DRMM <sub>ll</sub>	<b>0.56*</b>	<b>0.5244*</b>	<b>0.4773</b>	<b>0.424</b>	<b>0.2989</b>	<b>0.4167</b>	<b>0.3722*</b>	<b>0.37</b>	<b>0.355*</b>	<b>0.2725*</b>
Listwise loss with sampling → DRMM <sub>ll-ws</sub>	0.4667	0.5289	0.4747	0.4227	0.2962	0.4	0.3722	0.3667	0.36	0.2634

More gain on ClueWeb than Robust04!

Better performance  
Especially on top-ranked results

Method	Robust04				ClueWeb			
	nDCG@1	nDCG@3	nDCG@5	nDCG@10	nDCG@1	nDCG@3	nDCG@5	nDCG@10
DRMM <sub>pl</sub>	0.4666	0.4422	0.4385	0.4252	0.3484	0.3118	0.32	0.3277
DRMM <sub>ll</sub>	<b>0.4711</b>	<b>0.4742*</b>	<b>0.4528</b>	<b>0.4355</b>	<b>0.3732</b>	<b>0.359*</b>	<b>0.3595*</b>	<b>0.3666*</b>
DRMM <sub>ll-ws</sub>	0.4133	0.4589	0.442	0.4283	0.3565	0.3539	0.3519	0.3638

## Experimental Setup

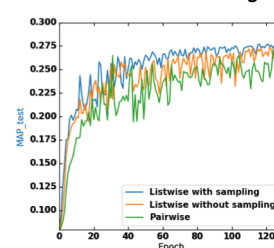
Datasets:

- Robust'04
  - 250 topics
- ClueWeb'09-Category-B
  - 200 topic
- Retrieved the top 2,000 documents per query
- pre-trained word embeddings
  - dimension 300 from Glove

## Listwise Loss Function

- Used the loss function of ListNet algorithm
- based on estimating a probability distribution for a list of scored documents
- Cross-entropy is used to measure the distance between the two probability distributions

## Performance on test set during training



Less fluctuation in MAP performance, when

- 1) Model is trained with listwise loss function
- 2) Train data is sampled in each epoch