Paper Breakdown

Separate and Attend in Personal Email Search

Yu Meng, Maryam Karimzadehgan, Honglei Zhuang, Donald Metzler

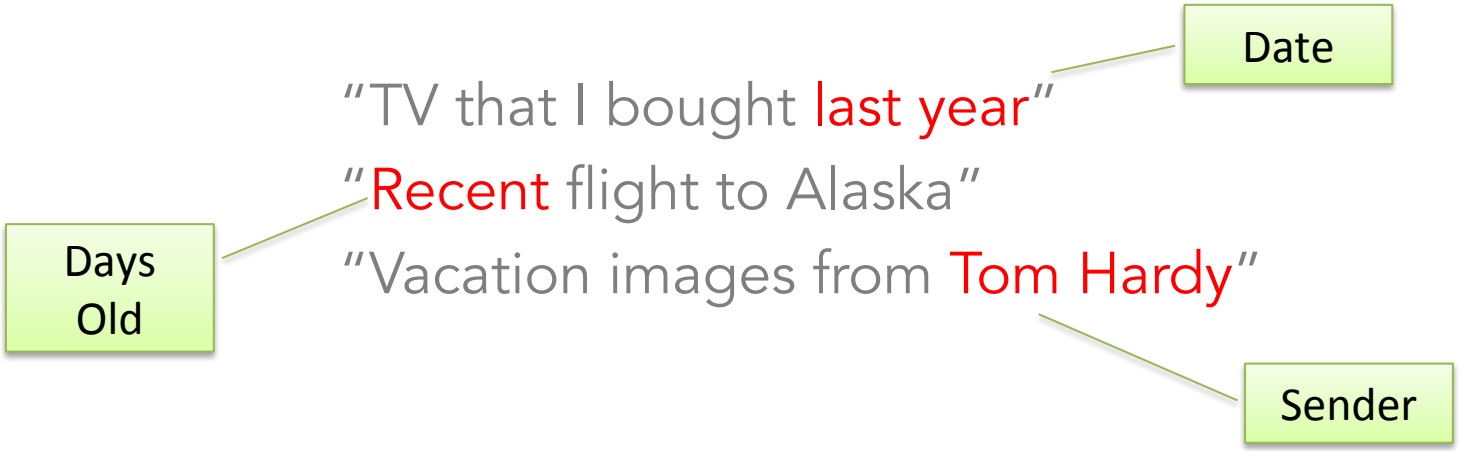
---

Rizu Jain

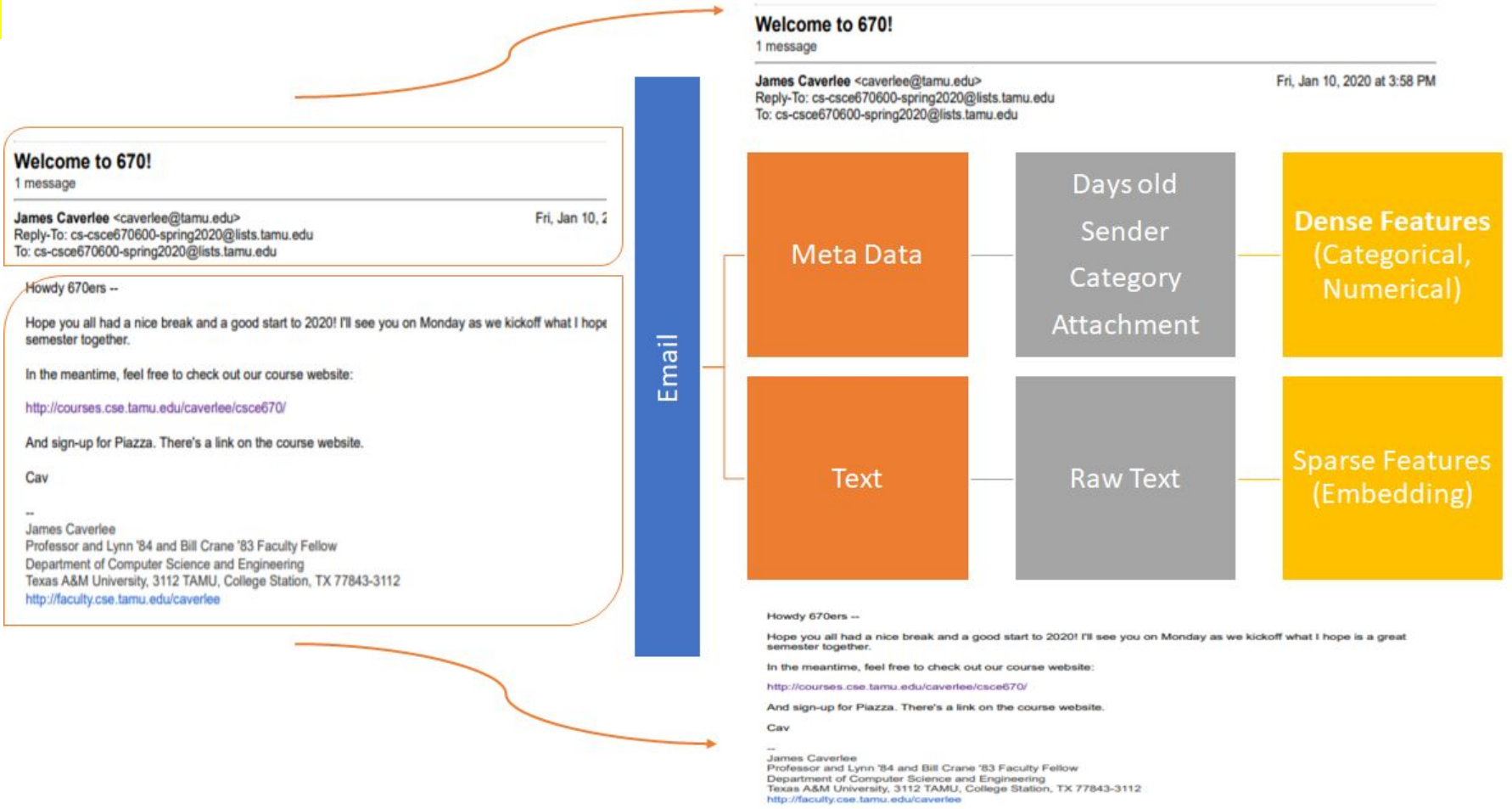
Motivation

"Search is not just query matching"

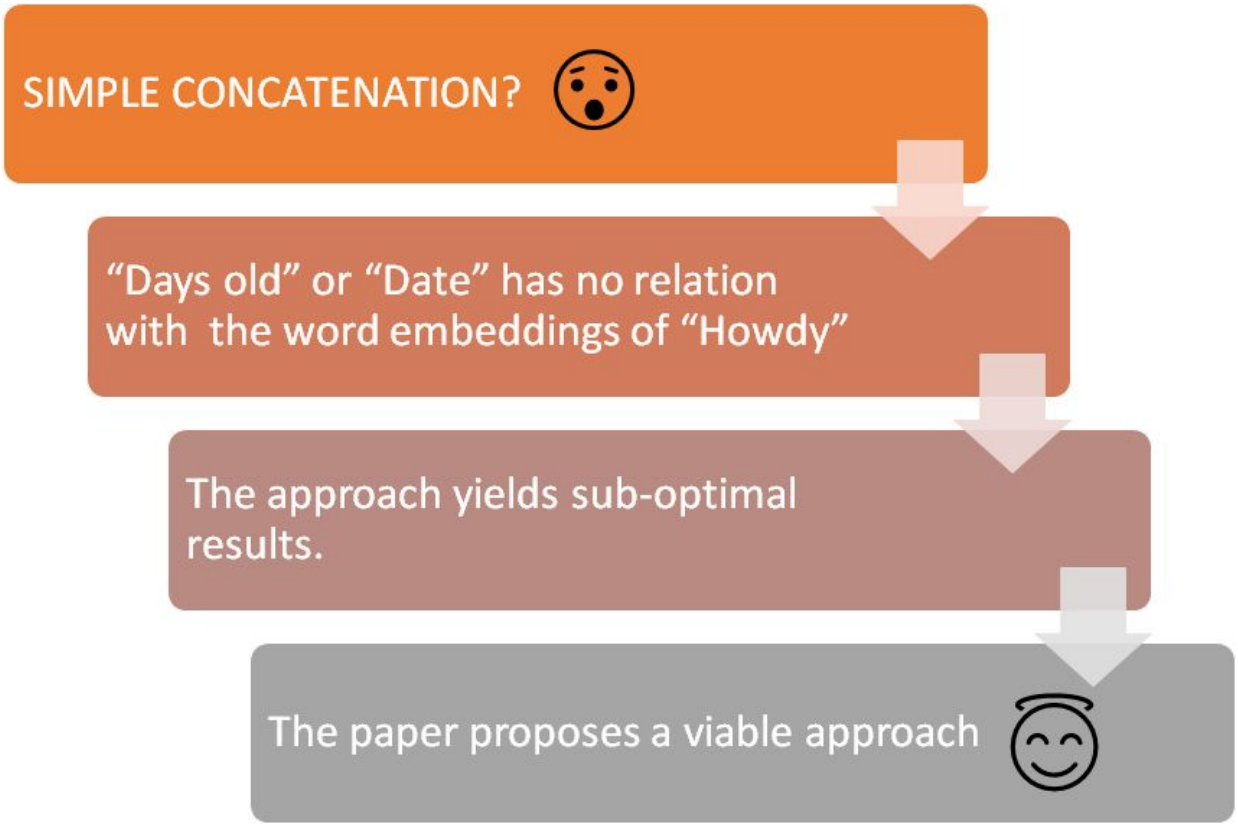
When searching for personal emails, following are plausible queries:



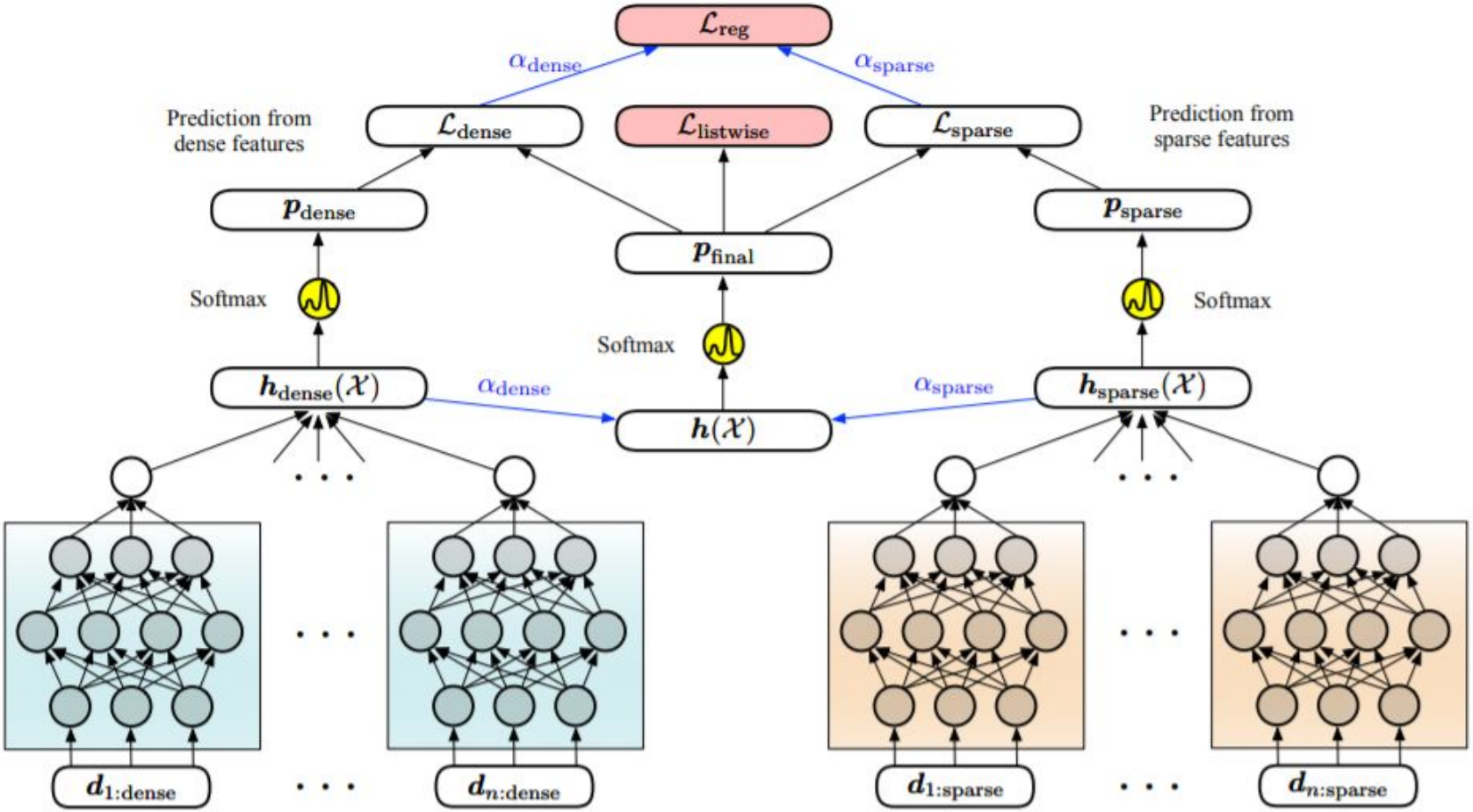
E-Mail: Sparse and Dense Data



How do we combine them?



Principal Contributions


$\mathcal{L}_{\text{reg}}$
α_{dense}
α_{sparse}
Prediction from dense features
$\mathcal{L}_{\text{dense}}$
$\mathcal{L}_{\text{listwise}}$
$\mathcal{L}_{\text{sparse}}$
Prediction from sparse features
$\boldsymbol{p}_{\text{dense}}$
$\boldsymbol{p}_{\text{final}}$
$\boldsymbol{p}_{\text{sparse}}$
Softmax
Softmax
Softmax
$\boldsymbol{h}_{\text{dense}}(\mathcal{X})$
α_{dense}
$\boldsymbol{h}(\mathcal{X})$
α_{sparse}
$\boldsymbol{h}_{\text{sparse}}(\mathcal{X})$
$\boldsymbol{d}_{1:\text{dense}}$
$\boldsymbol{d}_{n:\text{dense}}$
$\boldsymbol{d}_{1:\text{sparse}}$
$\boldsymbol{d}_{n:\text{sparse}}$

Approach

1

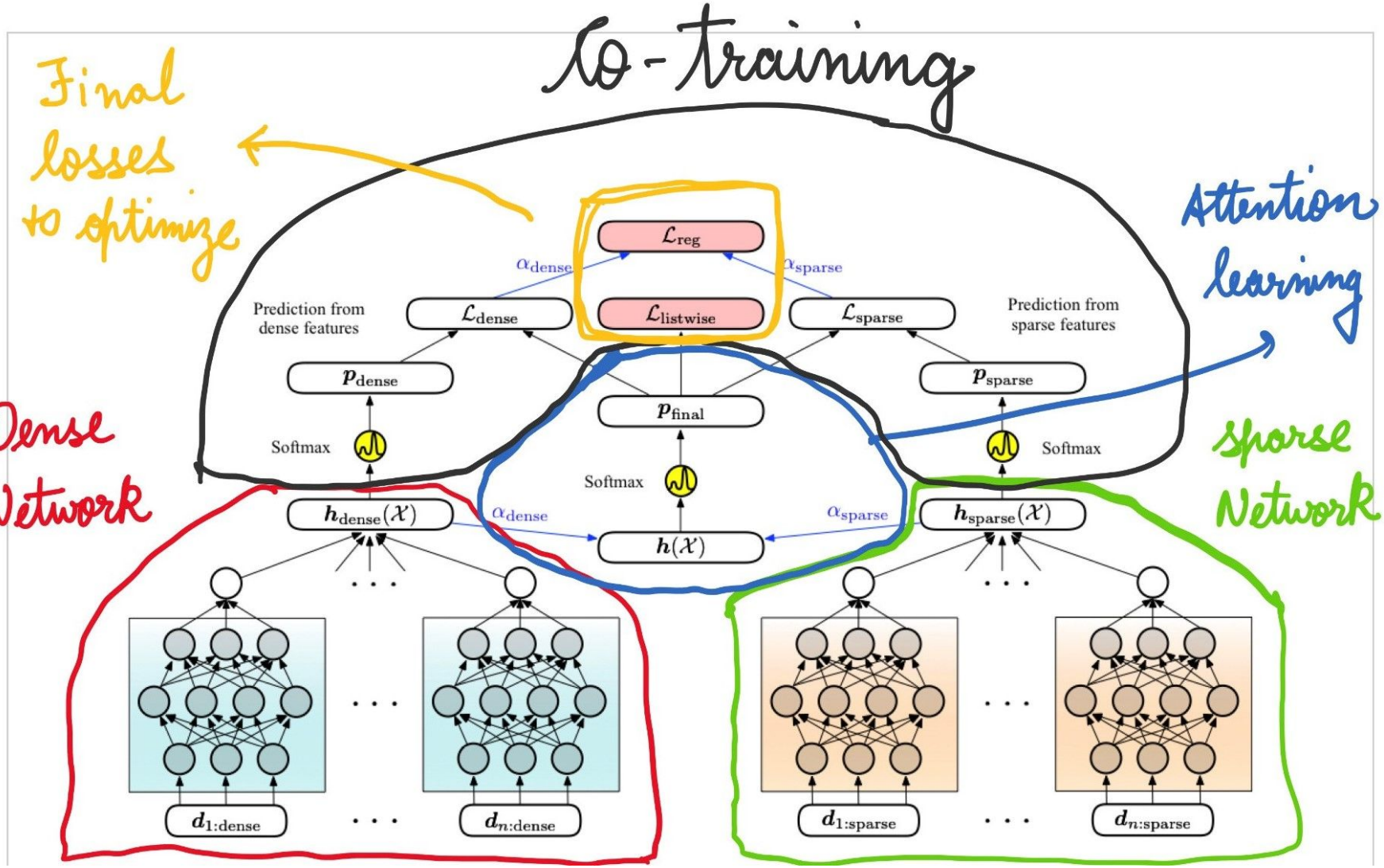
Train 2 separate neural networks

2

Learn attention weights from these 2 to get final predictions

3

Use co-training to improve individual networks thereby improving overall performance.


Co-training
Final losses to optimize
Attention learning
Dense Network
Sparse Network
$\mathcal{L}_{\text{reg}}$
α_{dense}
α_{sparse}
$\mathcal{L}_{\text{dense}}$
$\mathcal{L}_{\text{listwise}}$
$\mathcal{L}_{\text{sparse}}$
Prediction from dense features
Prediction from sparse features
p_{dense}
p_{final}
p_{sparse}
Softmax
Softmax
Softmax
$h_{\text{dense}}(\mathcal{X})$
$h(\mathcal{X})$
$h_{\text{sparse}}(\mathcal{X})$
$d_{1:\text{dense}}$
$d_{n:\text{dense}}$
$d_{1:\text{sparse}}$
$d_{n:\text{sparse}}$

Co-Training

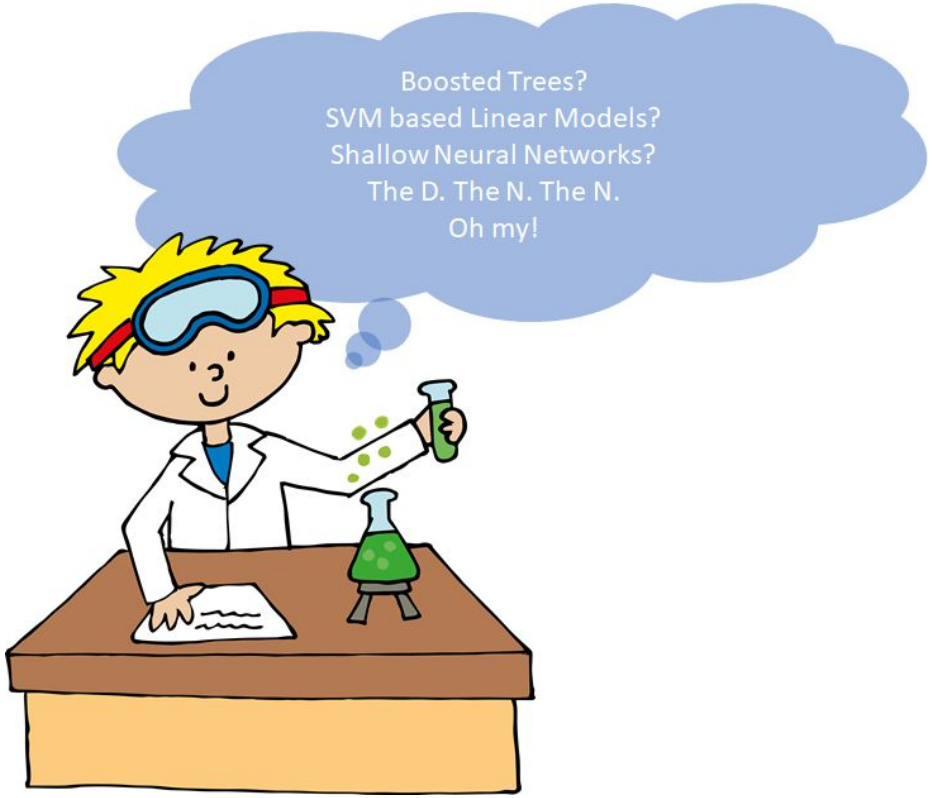
Motivation

"When two learning models capture different and complementary feature sets of the same instances, they can mutually enhance each other."

Challenge

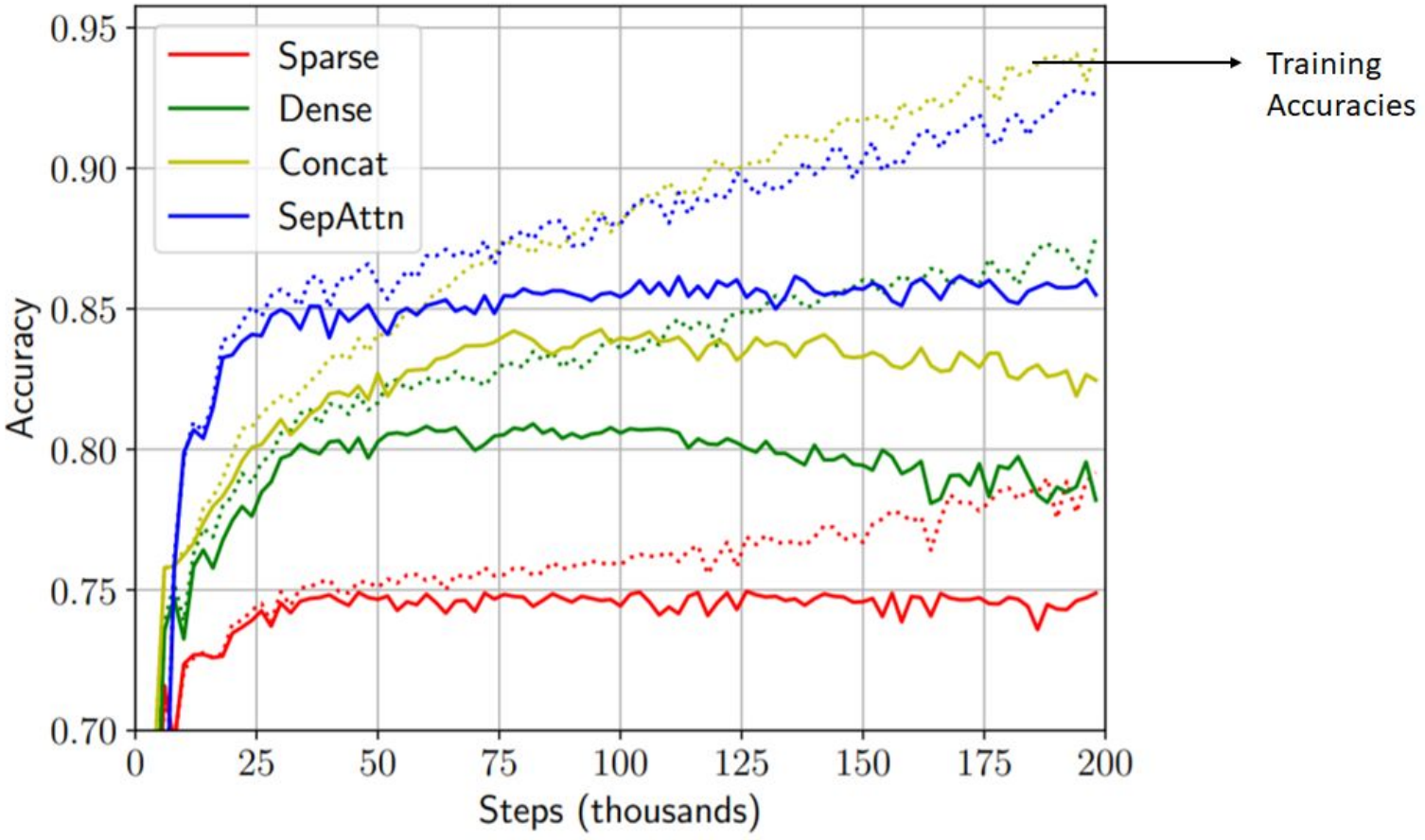
"The scales of the hidden states of those two models are very different."

1. Apply Softmax to convert the hidden states of those 2 layers into probability distribution
2. Compute KL divergence with the results obtained by merging the 2 networks
3. Do a weighted average to get a “regularization” loss which will motivate individual models to improve predictions



Experimental Findings

Reduced Overfitting

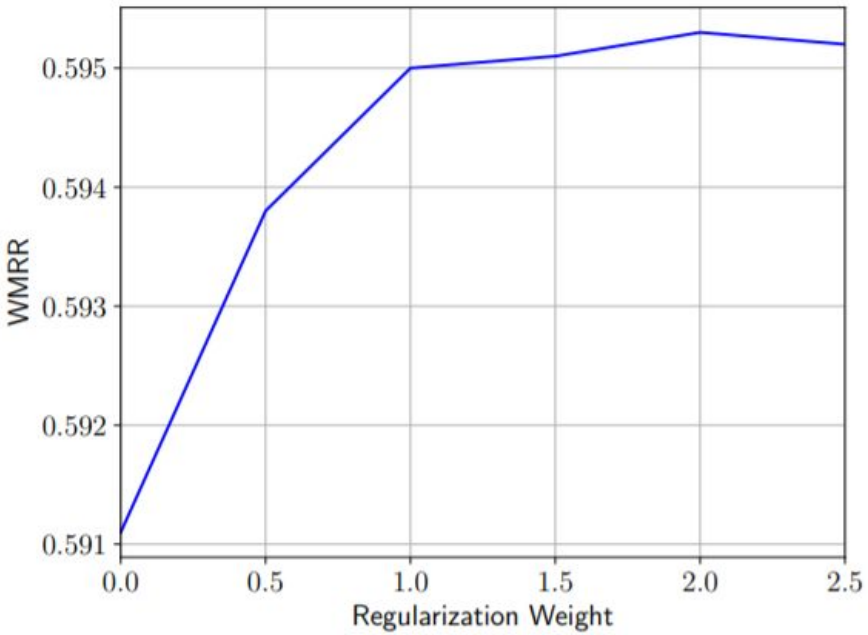


Better Recommendations

The data we use comes from the search click logs of Gmail search engine. The training set contains around 317 million queries, and the testing set contains around 41 million queries.

Methods	MRR (↑)	WMRR (↑)	ARP (↓)	WARP (↓)	DCG (↑)
Dense-only method	0.650	0.563	2.143	2.474	0.759
Sparse-only method	0.665	0.578	2.067	2.412	0.767
Concatenation method	0.682	0.589	2.002	2.366	0.770
SepAttn	**0.686**	**0.595**	**1.992**	**2.350**	**0.781**
Δ(%)	+0.59*	+1.02*	−0.50*	−0.68*	+1.43*

Co-training helps ...



Regularization weight is the multiplication factor for the "regularization loss" obtained for co-training

My Thoughts

Limitations

Datasets for personal email recommendation are hard to find because of privacy concerns:

Available datasets also have to be anonymized before using which limits the model architecture that could be used.

Future Work

Enablement of ML on edge devices could facilitate evaluation of learning algorithms without compromising privacy.

Thereby rich architectures could be explored for personal email search too.

Deploy to other search domains

Photo Gallery

" Last christmas photos"

Social Media Posts

" Megan's Vegas photos"

There is much scope of improvement in search experiences by leveraging information to make the inferences smarter.

On second thoughts,
We should just wait for Jeff Dean to release his omni-modal embeddings.