The University of Texas at Austin

UCLA

# End-to-End Learning to Index and Search in Large Output Spaces

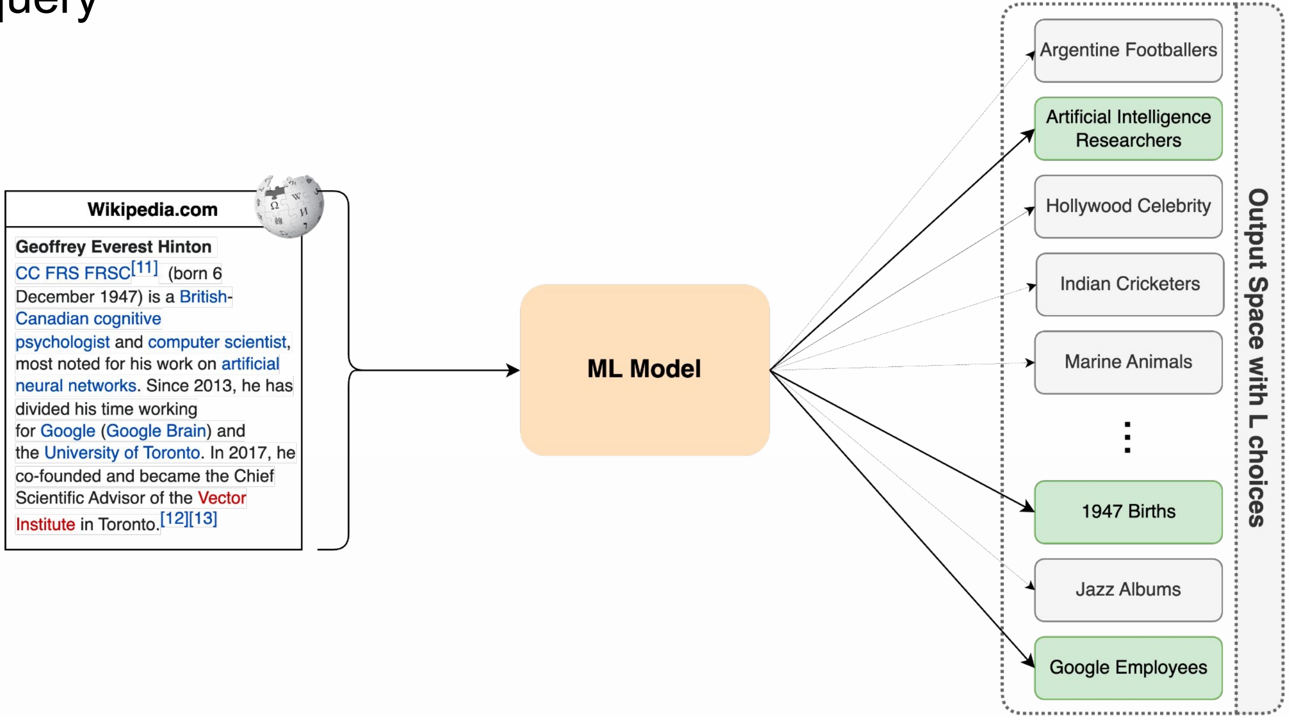
Nilesh Gupta, Patrick H. Chen, Hsiang-Fu Yu, Cho-Jui Hsieh, Inderjit S. Dhillon

**Nilesh Gupta**
PhD Student, The University of Texas at Austin

NEURAL INFORMATION PROCESSING SYSTEMS

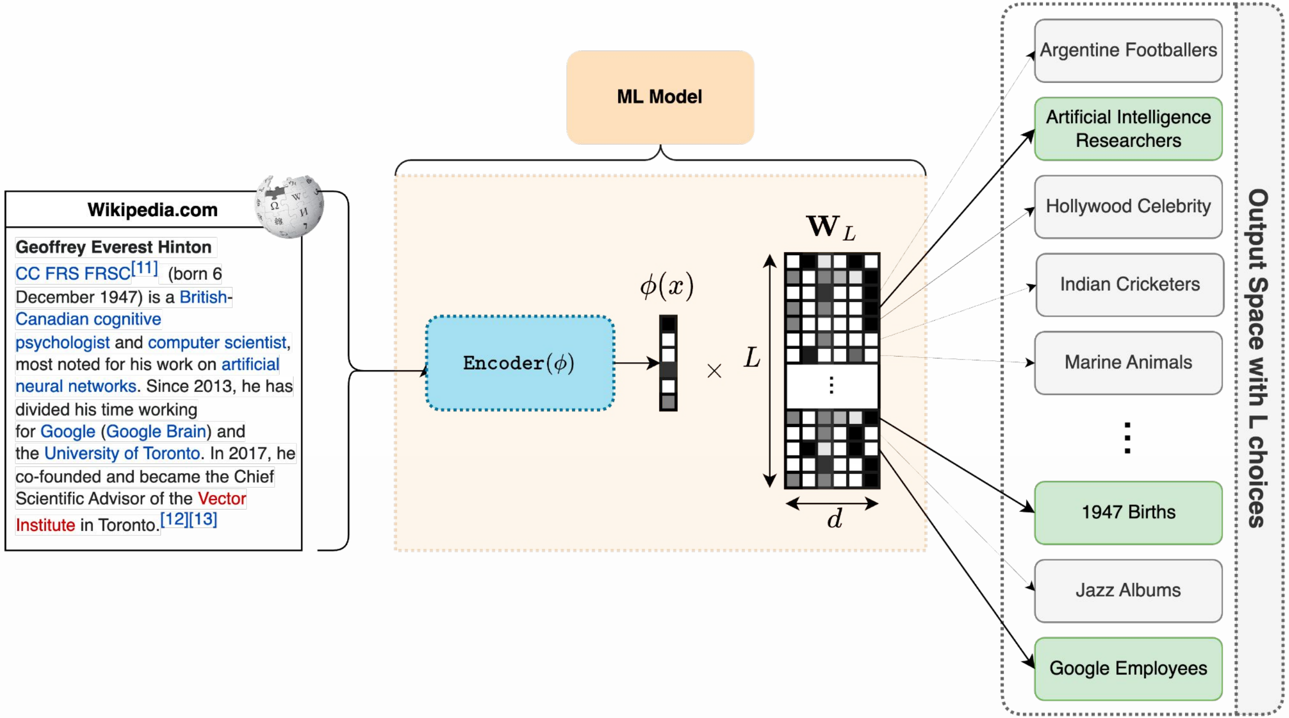
# Introduction

- Multi-label classification - predict set of all relevant labels (output choices) for a query



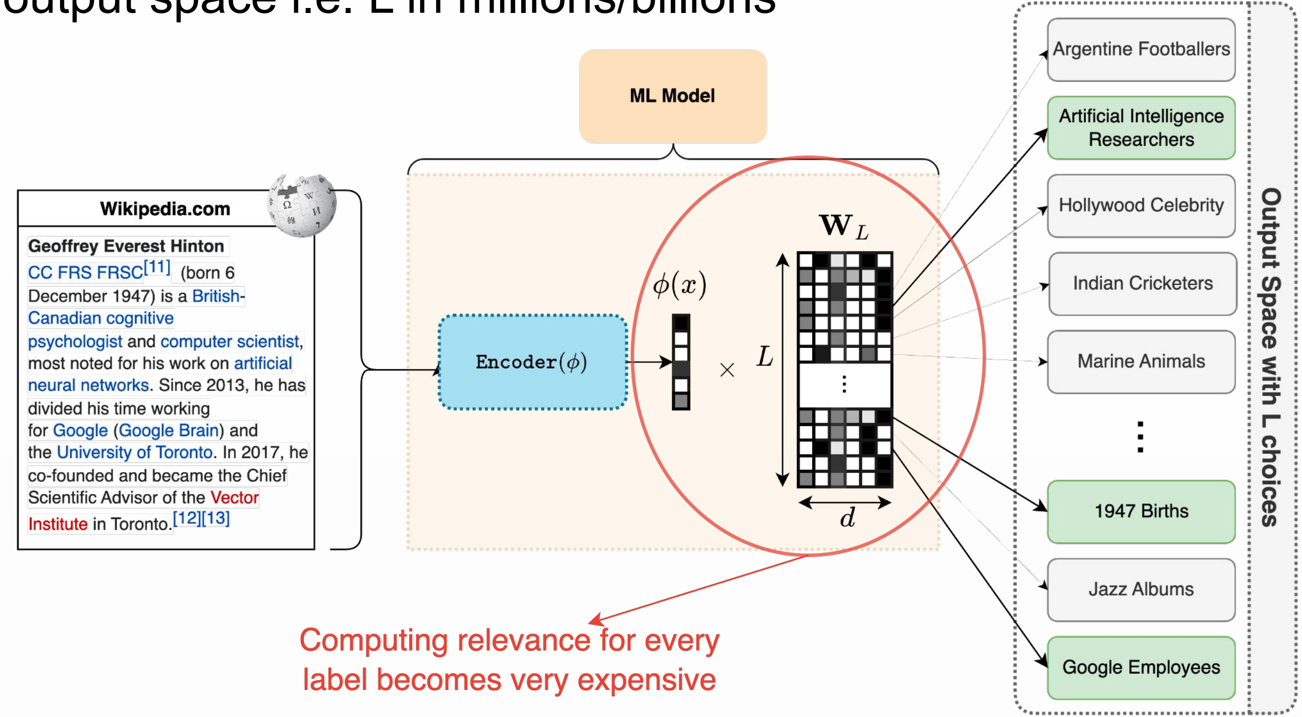
# Introduction

- Typical approach to solve multi-label classification



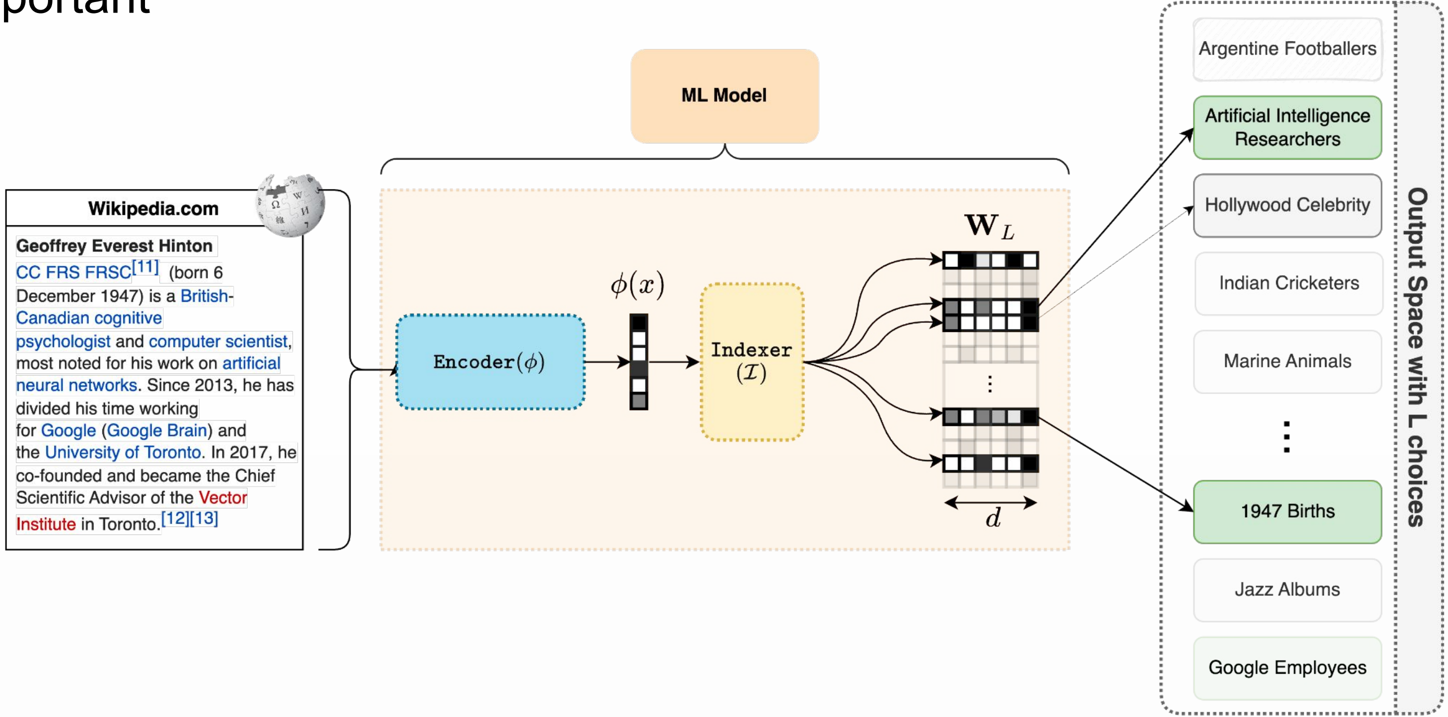
# Introduction

- Many real-world scenarios (recommendation, openQA, etc) have very large output space i.e. L in millions/billions



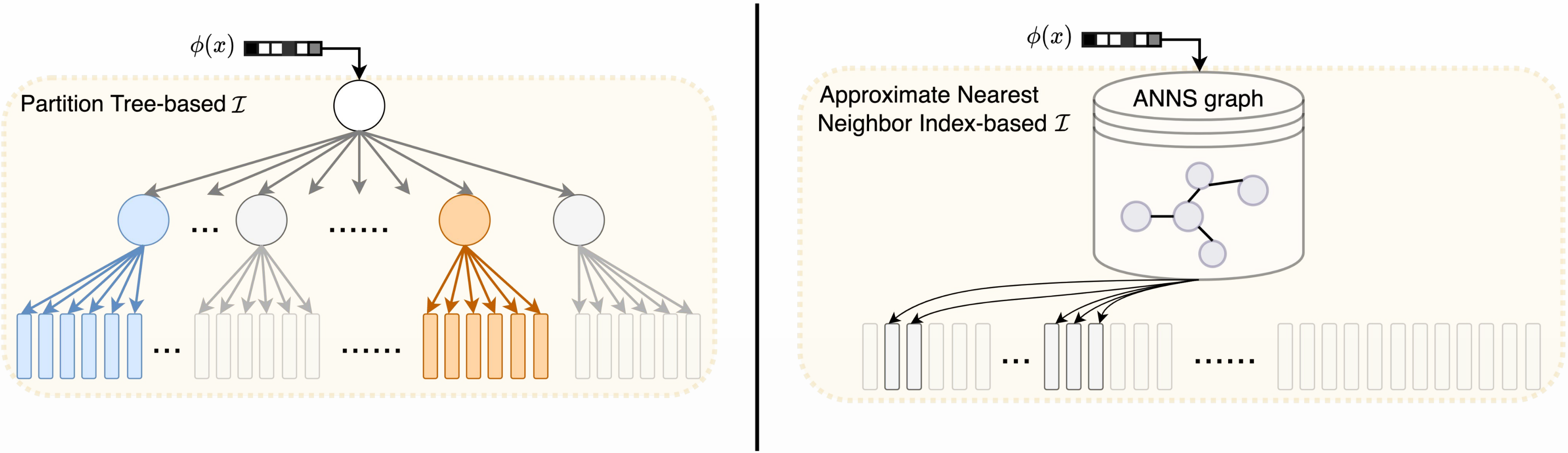
# Existing Approaches

- Indexer $\mathcal{I}$ efficiently samples only a few label indices, quality of $\mathcal{I}$ is important



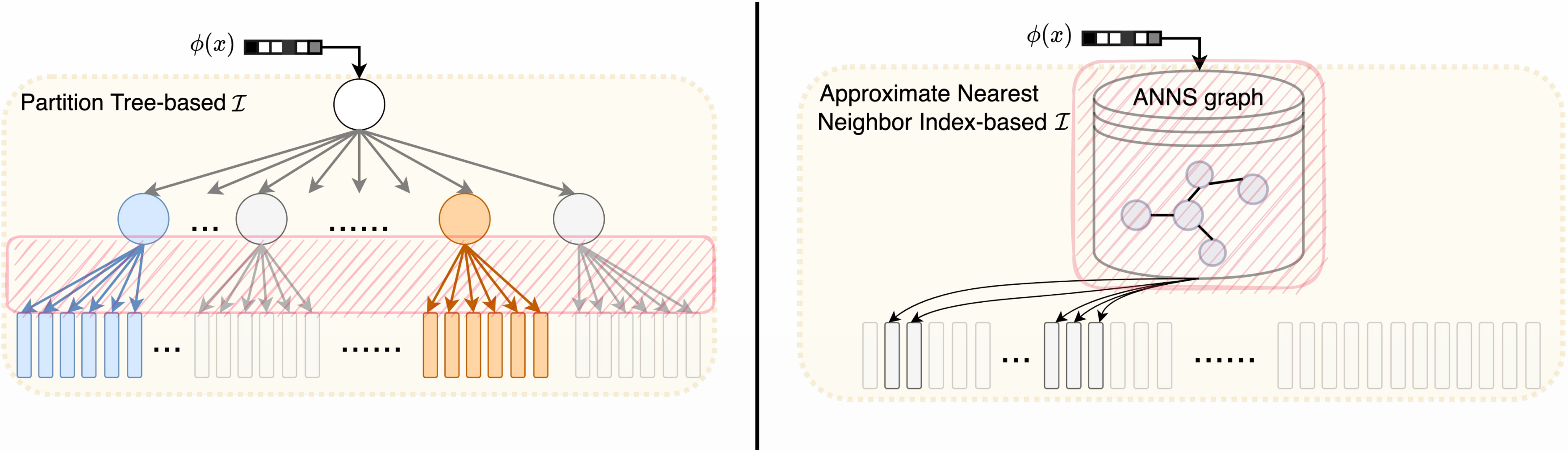
# Existing Approaches

- Popular choices for search index - partition tree-based and ANNS-based



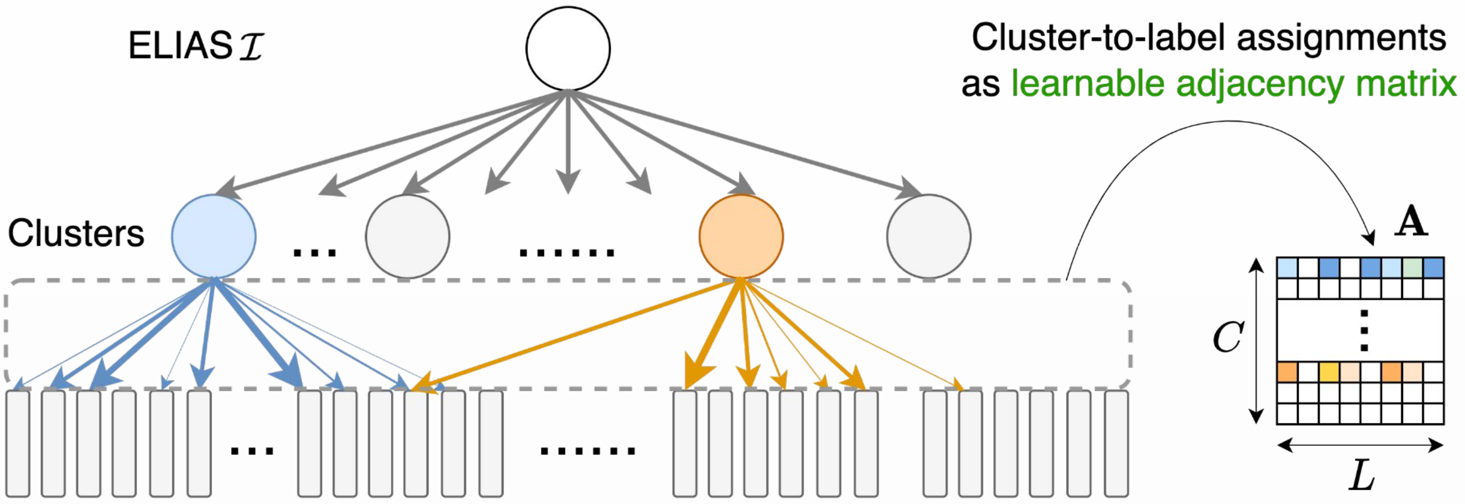
# Existing Approaches

- Both of these approaches fix their index structure before training
- Search performance limited to the quality of choices made during initialization



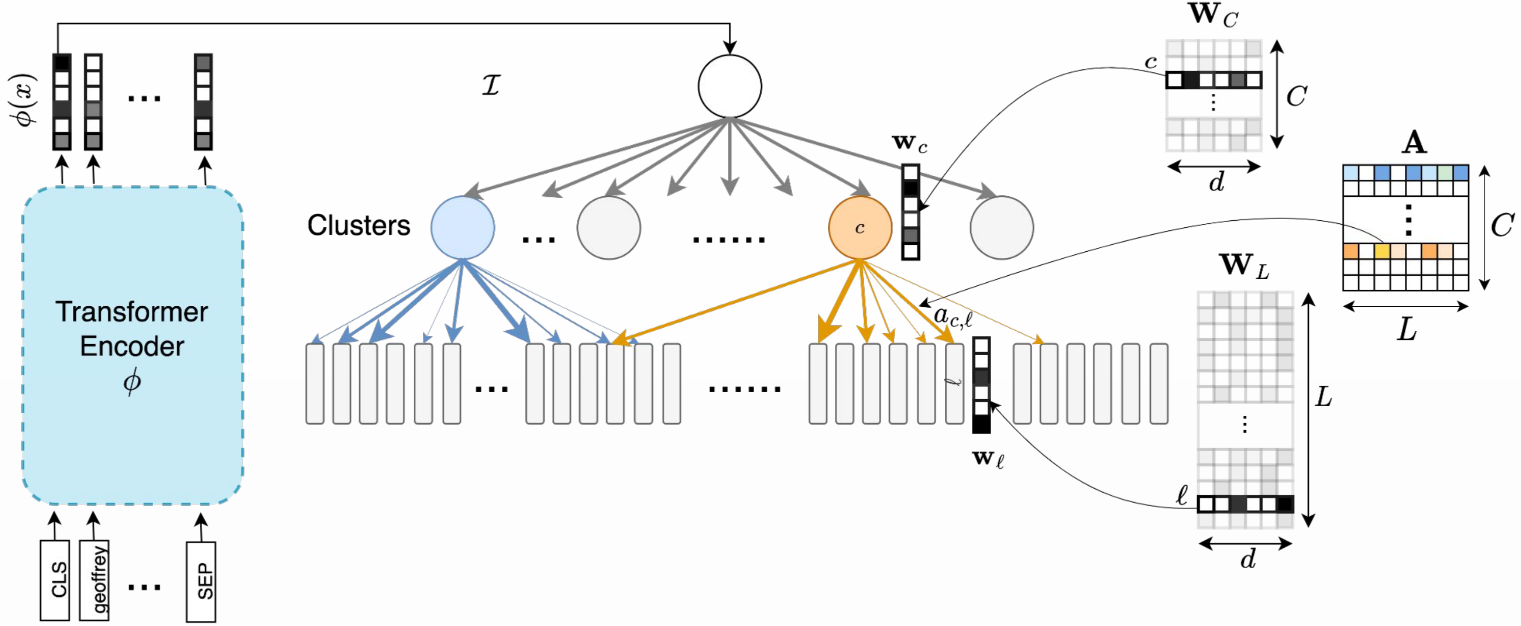
# ELIAS

- Relaxes partition tree-based index to weighted graph-based index
- Parameterize cluster-to-label edges as learnable adjacency matrix
- Learn A end-to-end with rest of the model parameters (encoder, classifiers)

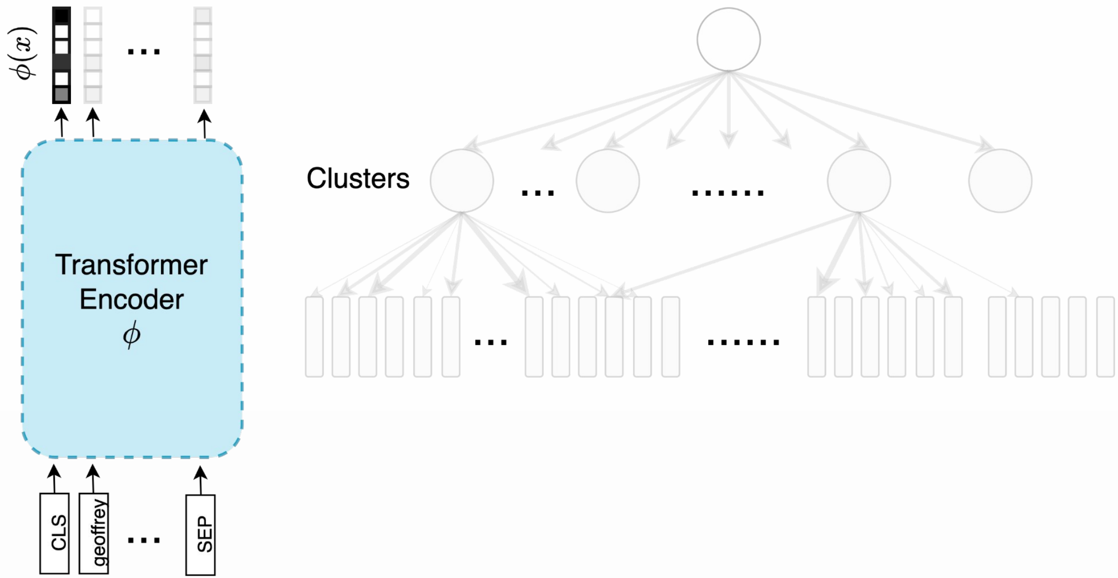


# ELIAS model

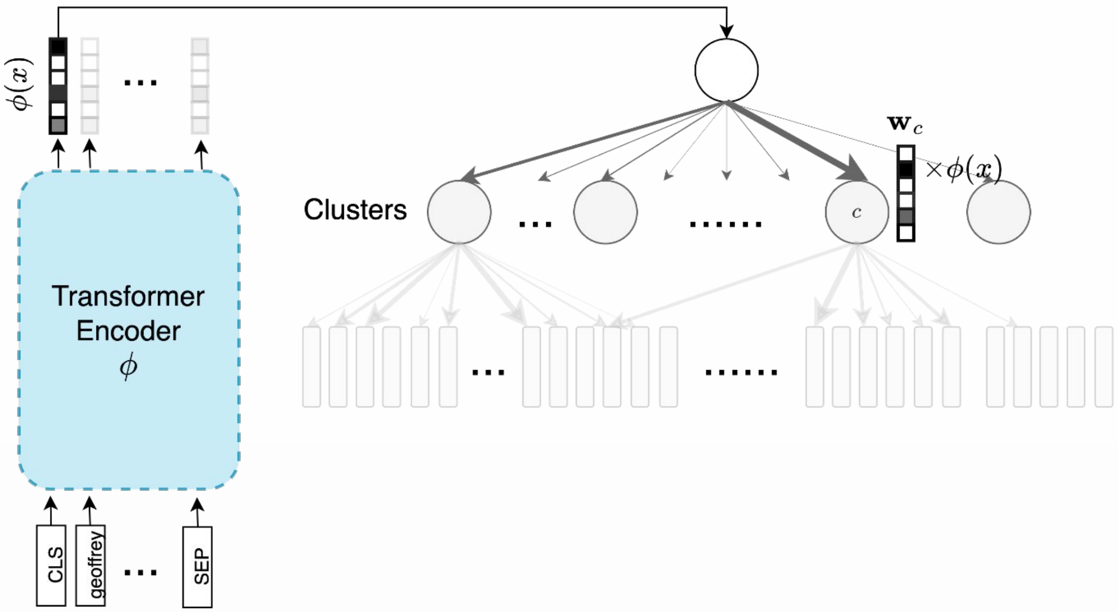
- Model parameters - $\phi, \mathbf{W}_C, \mathbf{A}, \mathbf{W}_L$



# ELIAS forward

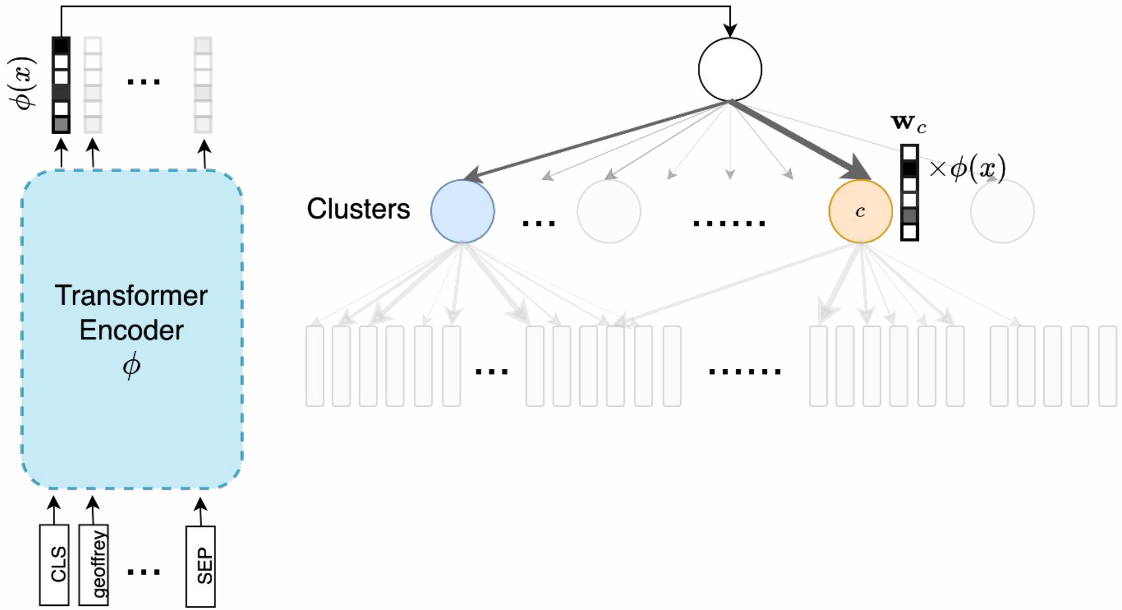


# ELIAS forward




| Step 1 |
| --- |
| Score all clusters $\longrightarrow \hat{s}_{\mathcal{C}} \sim \mathbf{W}_{C} \times \phi(x)$ |

# ELIAS forward



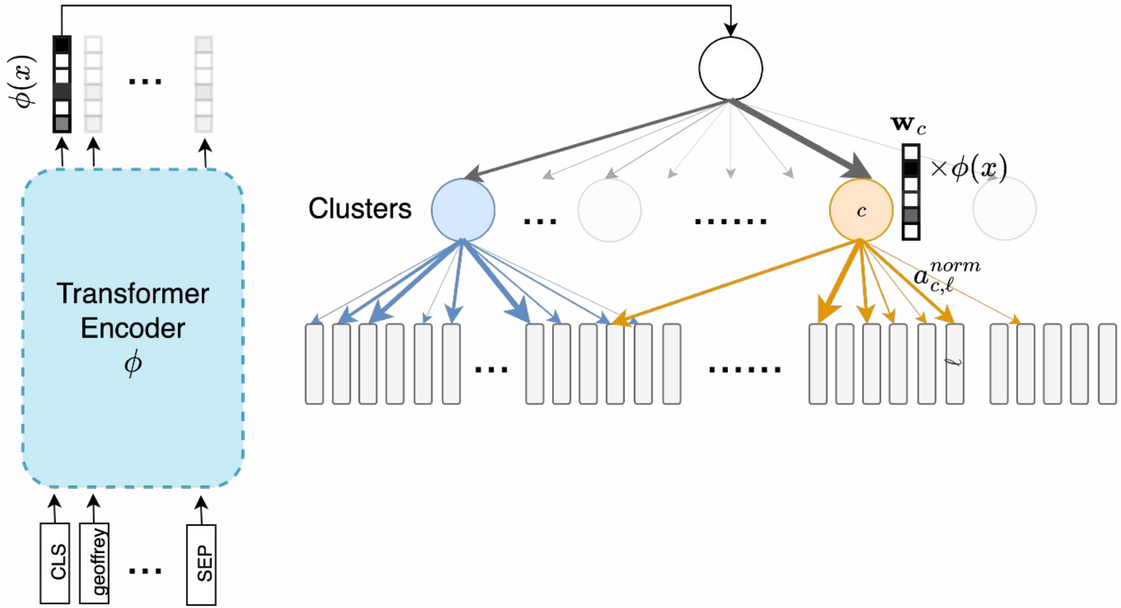

**Step 1**

Score all clusters $\longrightarrow \hat{s}_{\mathcal{C}} \sim \mathbf{W}_{\mathcal{C}} \times \phi(x)$

**Step 2**

Select top clusters $\longrightarrow \text{argtop-}b(\hat{s}_{\mathcal{C}})$

# ELIAS forward

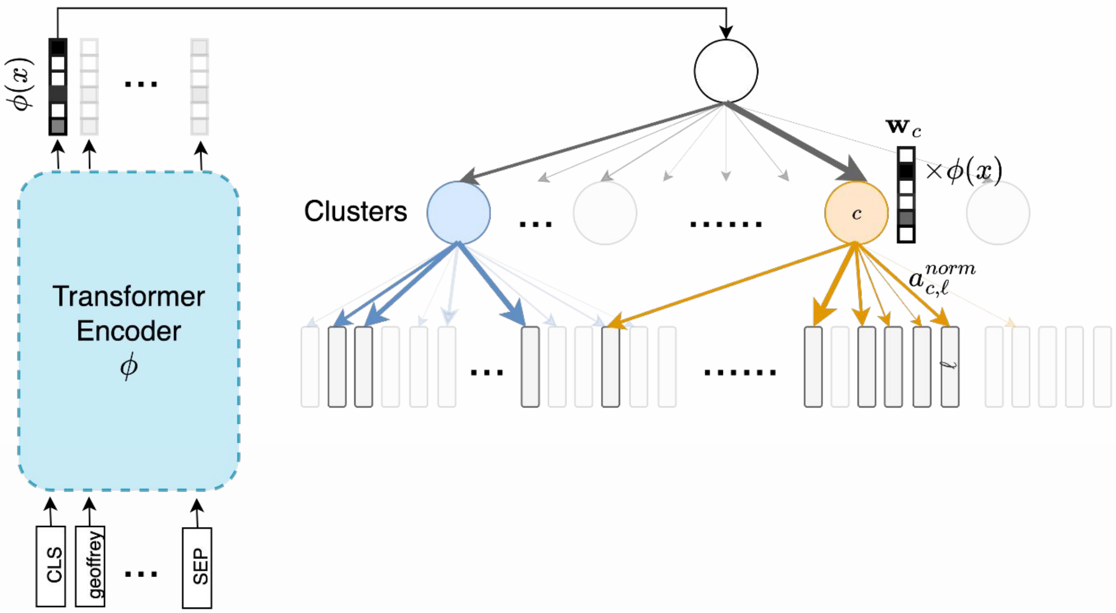



| Step 1 |
|---|
| Score all clusters $\longrightarrow \hat{s}_{\mathcal{C}} \sim \mathbf{W}_{\mathcal{C}} \times \phi(x)$ |

| Step 2 |
|---|
| Select top clusters $\longrightarrow \text{argtop-}b(\hat{s}_{\mathcal{C}})$ |

| Step 3 |
|---|
| Score all potential paths through top clusters $\hat{s}_c * a_{c,l}^{norm}$ |

# ELIAS forward



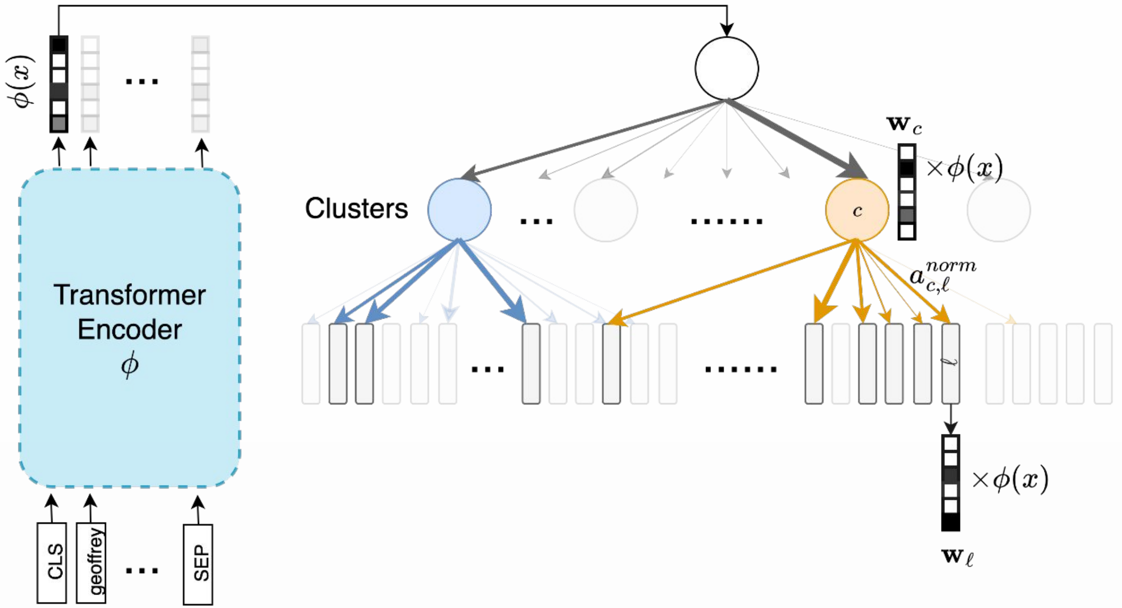

| Step 1 |
| --- |
| Score all clusters $\longrightarrow \hat{s}_{\mathcal{C}} \sim \mathbf{W}_C \times \phi(x)$ |

| Step 2 |
| --- |
| Select top clusters $\longrightarrow \text{argtop-}b(\hat{s}_{\mathcal{C}})$ |

| Step 3 |
| --- |
| Score all potential paths through top clusters $\hat{s}_c * a_{c,l}^{norm}$ |

| Step 4 |
| --- |
| Select top K labels based on path scores |

# ELIAS forward




**Step 1**

Score all clusters $\longrightarrow \hat{s}_{\mathcal{C}} \sim \mathbf{W}_{C} \times \phi(x)$

**Step 2**

Select top clusters $\longrightarrow \text{argtop-}b(\hat{s}_{\mathcal{C}})$

**Step 3**

Score all potential paths through top clusters

$\hat{s}_c * a_{c,l}^{norm}$

**Step 4**

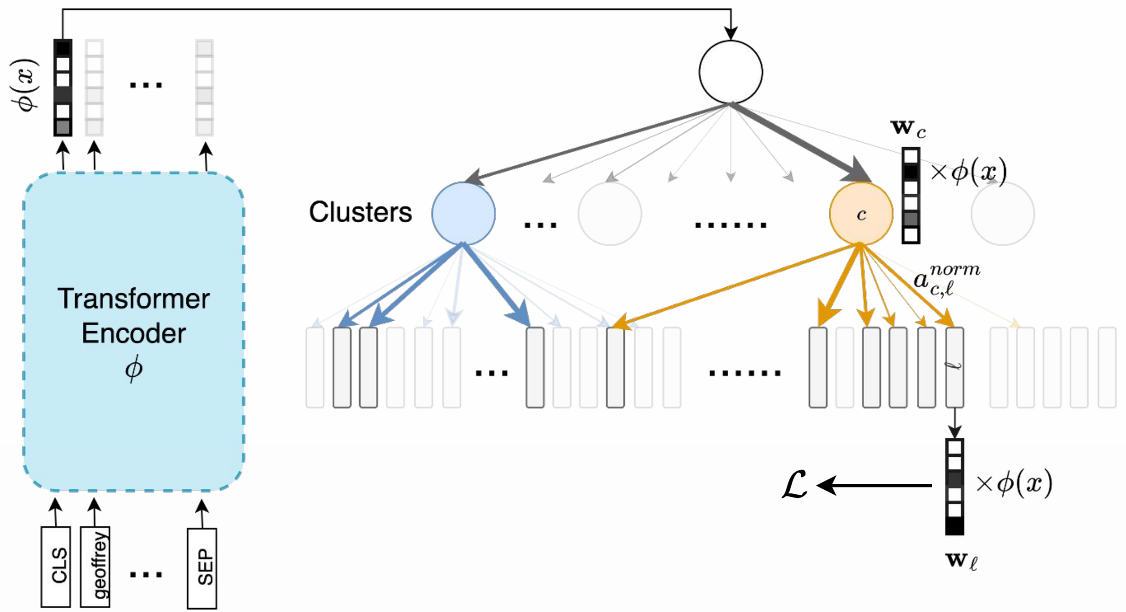
Select top K labels based on path scores

**Step 5**

Evaluate label classifier for all top K labels

Final score of $\ell$ : $\hat{s}_c * a_{c,l}^{norm} * \sigma(\mathbf{w}_\ell^T \phi(x))$

# ELIAS forward




**Step 1**

Score all clusters $\longrightarrow \hat{s}_\mathcal{C} \sim \mathbf{W}_C \times \phi(x)$

**Step 2**

Select top clusters $\longrightarrow \text{argtop-}b(\hat{s}_\mathcal{C})$

**Step 3**

Score all potential paths through top clusters

$\hat{s}_c * a_{c,l}^{norm}$

**Step 4**

Select top K labels based on path scores

**Step 5**

Evaluate label classifier for all top K labels

Final score of $\ell$ : $\hat{s}_c * a_{c,l}^{norm} * \sigma(\mathbf{w}_\ell^T \phi(x))$

# ELIAS optimization

- **Computational challenge** - operating on full adjacency matrix can be very expensive for web-scale datasets

- **Optimization challenge** - because of flexibility in the model to assign a label to various clusters, it becomes hard for a label to get confidently assigned to only a few relevant clusters

# ELIAS optimization

- **Computational challenge** - operating on full adjacency matrix can be very expensive for web-scale datasets

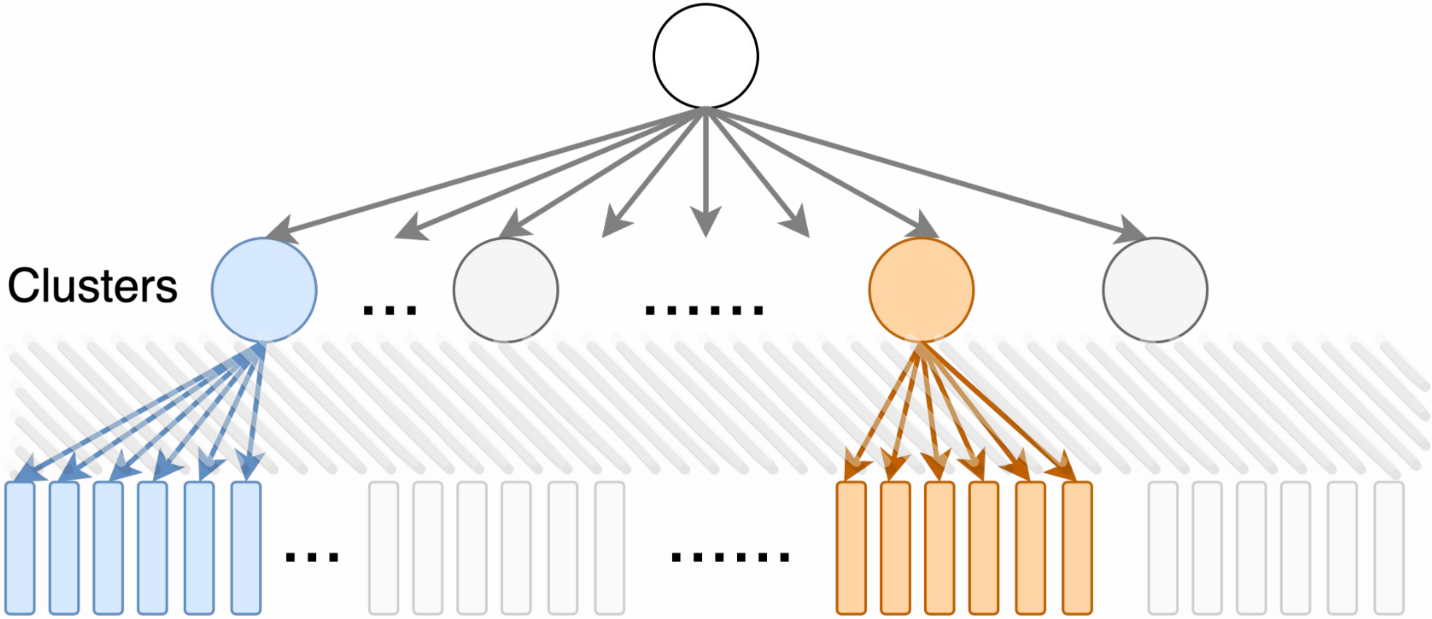
- Learn a row-wise sparse adjacency matrix

- **Optimization challenge** - because of flexibility in the model to assign a label to various clusters, it becomes hard for a label to get confidently assigned to only a few relevant clusters

- Train in two stages

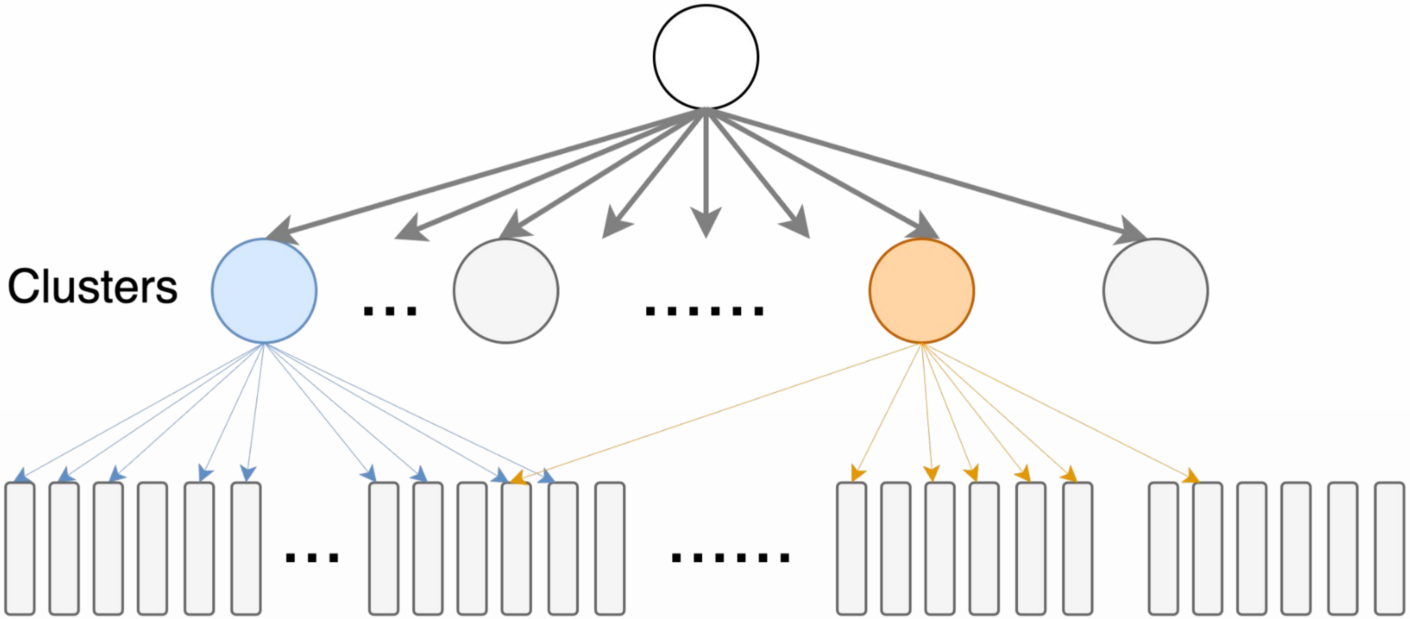
# ELIAS staged training

- **Stage 1:** fix $A$ as traditional partition clusters and train $\phi, \mathbf{W}_C, \mathbf{W}_L$



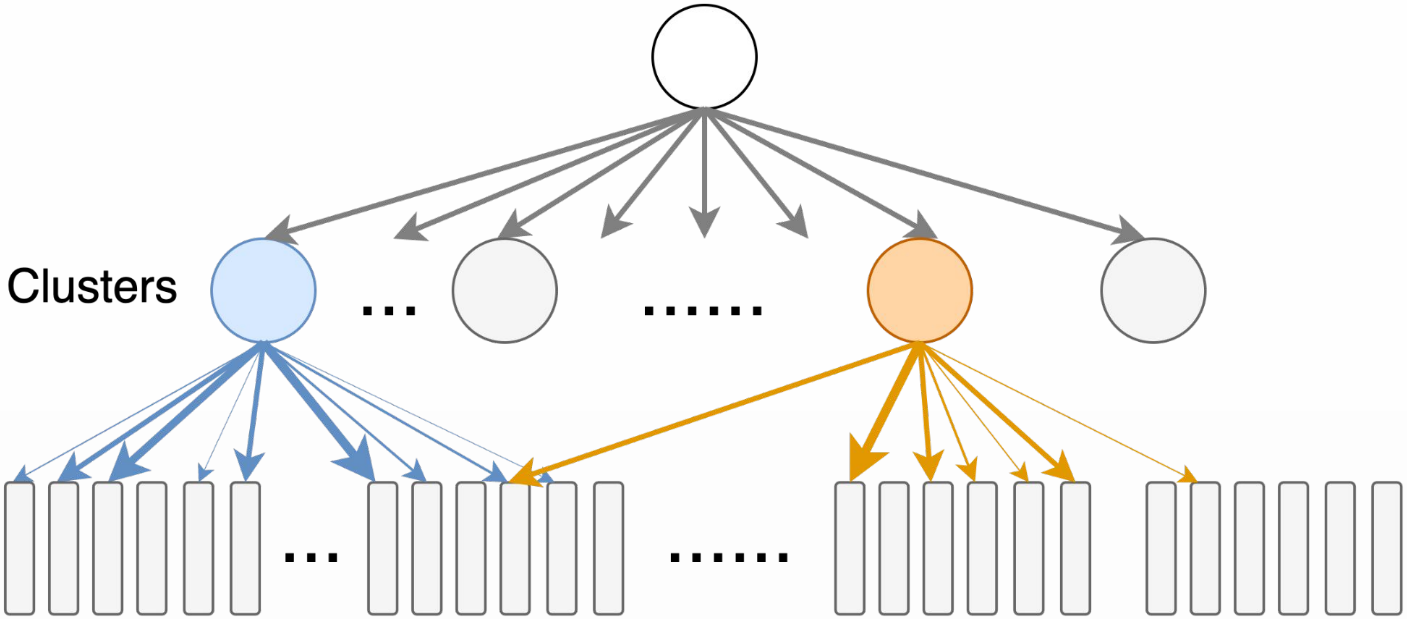
# ELIAS staged training

- Initialize an approximate row-wise sparse $A$ based on weighted count of number of times cluster $c$ gets assigned to positives of label $\ell$ by stage 1



# ELIAS staged training

- **Stage 2:** train full model i.e. $\phi, \mathbf{W}_C, \mathbf{W}_L$, and non-zero entries of $A$



# Experiments

- State-of-the-art on several large-scale extreme classification benchmarks

**Amazon-670K**

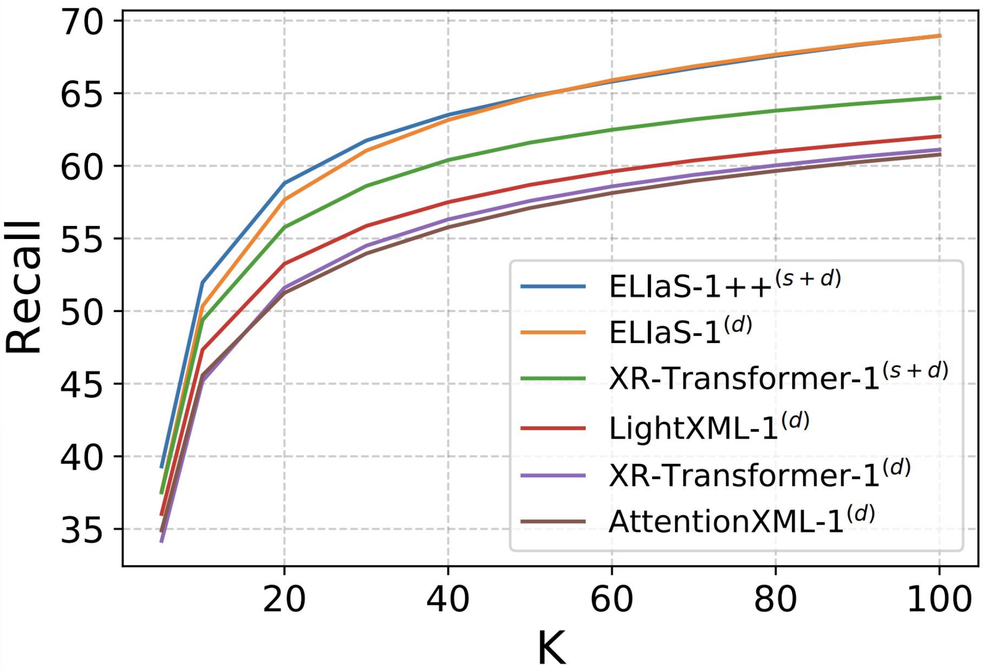
| Method | P@1 | P@3 | P@5 |
|---|---|---|---|
| **AttentionXML** | 47.58 | 42.61 | 38.92 |
| **LightXML** | 49.10 | 43.83 | 39.85 |
| **XR-Transformer** | 50.11 | 44.56 | 40.64 |
| **ELIAS** | 50.63 | 45.49 | 41.60 |
| **ELIAS++** | **53.02** | **47.18** | **42.97** |

**Wikipedia-500K**

| Method | P@1 | P@3 | P@5 |
|---|---|---|---|
| **AttentionXML** | 76.95 | 58.42 | 46.14 |
| **LightXML** | 77.78 | 58.85 | 45.57 |
| **XR-Transformer** | 79.40 | 59.02 | 46.25 |
| **ELIAS** | 79.00 | 60.37 | 46.87 |
| **ELIAS++** | **81.26** | **62.51** | **48.82** |

# Experiments

- Upto 4% better at R@100 than the next best method on Amazon-670K



# Thank You!

- **Paper** - https://arxiv.org/pdf/2210.08410.pdf
- **Code** - https://github.com/nilesh2797/ELIAS
- **Reach out** - nilesh@cs.utexas.edu