Weakly-Supervised Hierarchical Text Classification

Yu Meng, Jiaming Shen, Chao Zhang and Jiawei Han

Outline

- Preliminaries: Problem Formulation
- Methodology
- Experiment Results
- Case Studies

Problem Formulation

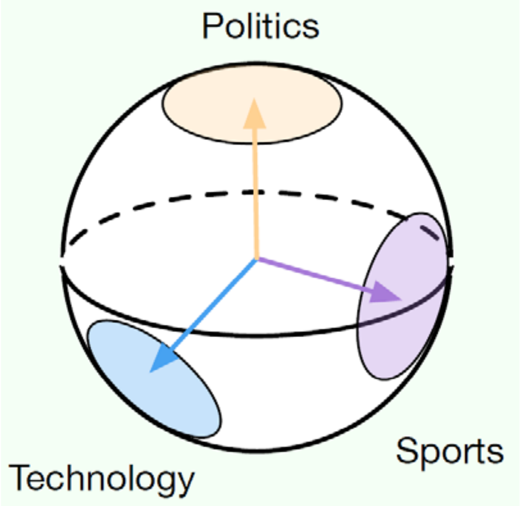
- Given a **text collection** and a **class hierarchy**, the task aims to assign each document the most appropriate class label;
- Consider tree-structured class categories;
- User provides **weak supervision** for each **leaf** class
  - Word-level; e.g. {“basketball”, “football”, ”tennis”}
  - Document-level: very few labeled documents (3-10 docs per class).
- The weak supervision sources of each **internal** class are an aggregation of those of all its descendant leaf classes;
- Documents can be assigned to both internal and leaf categories in the hierarchy.

Outline

- Preliminaries: Problem Formulation
- Methodology
- Experiment Results
- Case Studies

Model Class Distribution

- We model class semantic on a unit sphere in R^p
  - Directional similarities between vectors are more effective in capturing semantic correlations;
  - Words are represented by normalized p-dimensional word2vec embedding;
  - Class semantic = a probability distribution over vector directions in R^p.



Model Class Distribution

- We need to take parent-child relationship in the class hierarchy into consideration.
  - For leaf classes, we model the class semantic as one vMF distribution;
  - For internal classes, we model the class semantic as mixture of vMF distribution, since the semantics of a parent class can be seen as a mixture of the semantics of its children classes.

Model Class Distribution

- Step 1 – Retrieve representative keywords:
  - If word-level supervision is given, we use the average of their embedding to retrieve top-t nearest words in the semantic space;
  - If document-level supervision is given, we use tf-idf weighting to retrieve top-t keywords from these labeled documents.
  - t is set to be the largest number that does not results in overlapping words across different classes.

Model Class Distribution

- Step 2 – Fitting mixture of vMF distribution
  - We define the probability distribution of a class as

$$f(\boldsymbol{x};\boldsymbol{\Theta}) = \sum_{h=1}^{m} \alpha_h \, c_p(\kappa_h) e^{\kappa_h \boldsymbol{\mu}_{\boldsymbol{h}}^T \boldsymbol{x}}$$

where $\boldsymbol{\Theta} = \{\alpha_1, \dots, \alpha_m, \boldsymbol{\mu}_{\boldsymbol{1}}, \dots, \boldsymbol{\mu}_{\boldsymbol{m}}, \kappa_1, \dots, \kappa_m\}$

Model Class Distribution

- Step 2 – Fitting mixture of vMF distribution (cont'd):
  - We use EM framework to find the parameters $\boldsymbol{\Theta}$.
  - E-step:
    - $p(z_i = h \mid \boldsymbol{x}_i, \Theta^{(t)}) = \frac{\alpha_h^{(t)} f_h(x_i; \boldsymbol{\mu}_h^{(t)}, \kappa_h^{(t)})}{\sum_{h'=1}^{m} \alpha_{h'}^{(t)} f_{h'}(x_i; \boldsymbol{\mu}_{h'}^{(t)}, \kappa_{h'}^{(t)})}$;
  - M-step:
    - $\alpha_h^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} p(z_i = h \mid \boldsymbol{x}_i, \Theta^{(t)})$;
    - $\boldsymbol{r}_h^{(t+1)} = \sum_{i=1}^{n} p(z_i = h \mid \boldsymbol{x}_i, \Theta^{(t)})\, \boldsymbol{x}_i$;
    - $\boldsymbol{\mu}_h^{(t+1)} = \frac{\boldsymbol{r}_h^{(t+1)}}{\left\| \boldsymbol{r}_h^{(t+1)} \right\|}$;
    - $\frac{I_{p/2}(\kappa_h^{(t+1)})}{I_{p/2-1}(\kappa_h^{(t+1)})} = \frac{\left\| \boldsymbol{r}_h^{(t+1)} \right\|}{\sum_{i=1}^{n} p(z_i = h \mid \boldsymbol{x}_i, \Theta^{(t)})}$.

Pseudo Document Generation

- Based on the class distribution $f(\boldsymbol{x}; \boldsymbol{\mu}, \kappa)$, we generate pseudo documents as pseudo training data.
- Procedure:
  - Train an LSTM language model on the entire corpus;
  - Sample an embedding vector $\boldsymbol{v}_0$ from $f(\boldsymbol{x}; \boldsymbol{\mu}, \kappa)$;
  - Use $\boldsymbol{w}_0$, the closest word to $\boldsymbol{v}_0$ in embedding space as the beginning word of the pseudo document;
  - Feed the current sequence to the LSTM language model to generate the next word and attach it to the current sequence recursively;
  - Since the beginning word of the pseudo document comes directly from the class distribution, it ensures the generated document is correlated to the corresponding class.

Pseudo Document Generation

- Some sample generated pseudo document snippets of class “politics” for The New York Times dataset:
  - abortion rights is often overlooked by the president’s 30-feb format of a moonjock period that offered him the rules to…
  - immigrants who had been headed to the united states in benghazi, libya, saying that mr. he making comments describing…
  - budget increases on oil supplies have grown more than a ezio of its 20 percent of energy spaces, producing plans by 1 billion…

Hierarchical Classification Model

- Local Classifier Per Internal Class
  - We construct a neural classifier (CNN or RNN) for each internal class with two or more children classes;
  - Intuitively, the local classifier aims to classify the documents assigned to parent class into the children classes for more fine-grained predictions;
  - For each document D_i, the output of the local classifier can be interpreted as a conditional probability

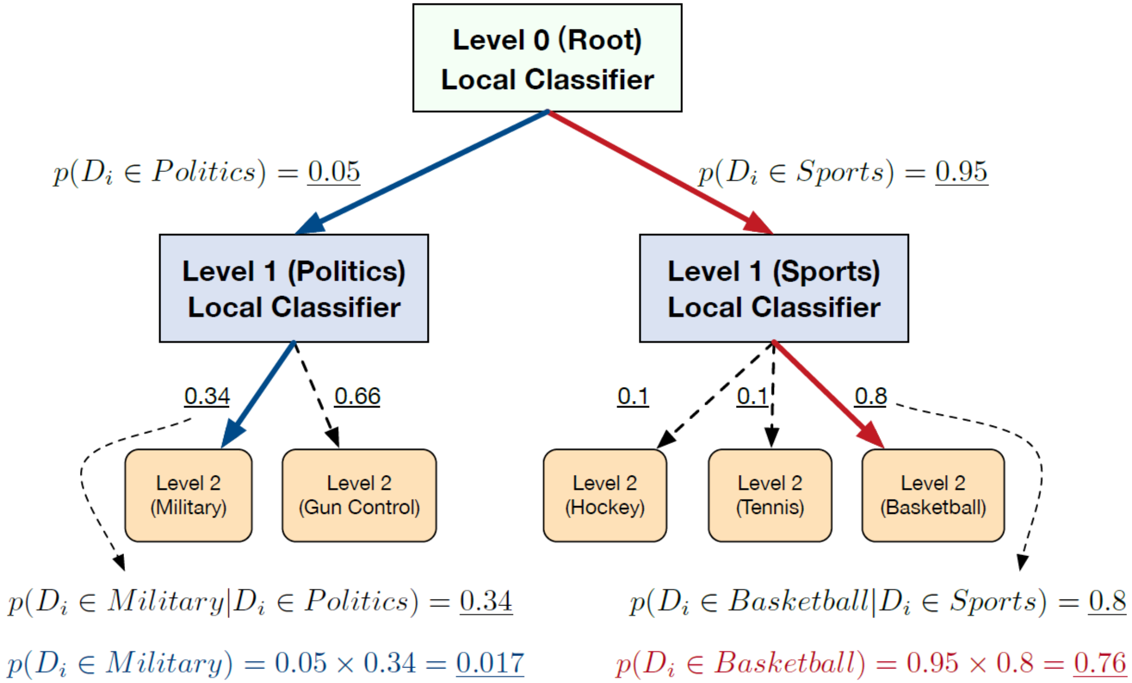
$$p(D_i \in C_{child} \mid D_i \in C_{parent})$$

Hierarchical Classification Model

- Local Classifier Pre-training
  - We generate β pseudo documents per class to pre-train the local classifier;
  - A naive way of creating the label for a pseudo document D_i^*:
    - Directly use the associated class label it is generated from; one-hot encodings;
    - Problem: classifier overfitting to pseudo documents.
  - Instead, use pseudo labels:
    - $l_{ij} = \begin{cases} (1-\alpha) + \alpha/m & D_i^* \text{ is generated from class } j \\ \alpha/m & \text{otherwise} \end{cases}.$
    - α accounts for the "noises" in pseudo documents; it is evenly split into all m classes.
  - Pre-training is performed by minimizing KL divergence loss to pseudo labels.

Hierarchical Classification Model

- Global Classifier Per Level
  - At each level k in the class taxonomy, we construct a global classifier by ensembling all local classifiers from root to level k.
  - Use unlabeled documents to bootstrap the global classifier.



Hierarchical Classification Model

- Global Classifier Construction
  - The multiplication operation can be explained by the conditional probability formula:

    $$p(D_i \in C_{child}) = p(D_i \in C_{child} \mid D_i \in C_{parent})p(D_i \in C_{parent})$$

  - All local classifiers from root to to level k are fine-tuned simultaneously via back-propagation during self-training; misclassifications at higher levels can be corrected.

Hierarchical Classification Model

- Global Classifier Self-training
  - Step 1: Use the pre-trained global classifier to classify all unlabeled documents in the corpus;
  - Step 2: Compute pseudo labels based on current predictions:

$l_{ij} = \frac{y_{ij}^2/f_j}{\sum_{j'} y_{ij'}^2/f_{j'}}$ where $f_j = \sum_i y_{ij}$ and y_{ij} is the current prediction.

  - Step 3: Minimize KL divergence loss to pseudo labels.
  - Iterate between Step 2 and 3 until less than $\delta\%$ of documents in the corpus have class assignment changes.

Hierarchical Classification Model

- Blocking Mechanism
  - Some documents should be classified into internal classes because they are more related to general topics rather than specific topics;
  - When a document D_i is classified into an internal class C_j, we use the output q of C_j's local classifier to determine whether or not D_i should be blocked at the current class:
    - If q is close to a one-hot vector, D_i should be classified into the corresponding child;
    - If q is close to uniform distribution, D_i should be blocked at current class;
    - Use normalized entropy as measure for blocking, i.e. block D_i if

$$-\frac{1}{\log m}\sum_{i=1}^{m} q_i \log q_i > \gamma$$

Hierarchical Classification Model

- Algorithm Summary

Algorithm 1: Overall Network Training.

Input: A text collection $\mathcal{D} = \{D_i\}|_{i=1}^{N}$; a class category tree $\mathcal{T}$; weak supervisions $\mathcal{W}$ of either $\mathcal{S}$ or $\mathcal{D}^L$ for each leaf class in $\mathcal{T}$.

Output: Class assignment $\mathcal{C} = \{(D_i, C_i)\}|_{i=1}^{N}$, where $C_i \in \mathcal{T}$ is the most specific class label for D_i.

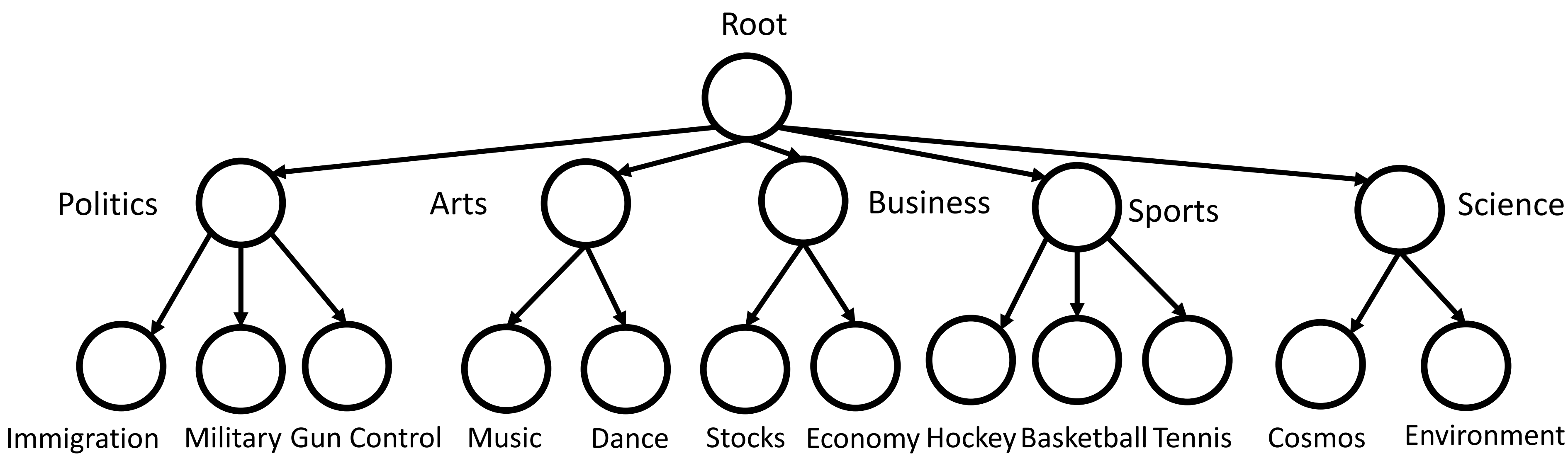
Initialize $\mathcal{C} \leftarrow \emptyset$;
for $k \leftarrow 0$ *to* $max\_level - 1$ **do**
    $\mathcal{N} \leftarrow$ all nodes at level k of $\mathcal{T}$;
    foreach $node \in \mathcal{N}$ **do**
        $\mathcal{D}^* \leftarrow$ Pseudo document generation;
        $\mathcal{L}^* \leftarrow$ Equation (1);
        pre-train $node.classifier$ with $\mathcal{D}^*, \mathcal{L}^*$;
    $G_k \leftarrow$ ensemble all classifiers from level 0 to k;
    while *not converged* **do**
        $\mathcal{L}^{**} \leftarrow$ Equation (2);
        self-train G_k with $\mathcal{D}, \mathcal{L}^{**}$;
    $\mathcal{D}_B \leftarrow$ documents blocked based on Equation (3);
    $\mathcal{C}_B \leftarrow \mathcal{D}_B$'s current class assignments;
    $\mathcal{C} \leftarrow \mathcal{C} \cup (\mathcal{D}_B, \mathcal{C}_B)$;
    $\mathcal{D} \leftarrow \mathcal{D} - \mathcal{D}_B$;
$\mathcal{C}' \leftarrow \mathcal{D}$'s current class assignments;
$\mathcal{C} \leftarrow \mathcal{C} \cup (\mathcal{D}, \mathcal{C}')$;
Return $\mathcal{C}$;

Outline

- Preliminaries: Problem Formulation
- Methodology
- Experiment Results
- Case Studies

Concrete Example: NYT

- Class Hierarchy (trimmed):



- Weak Supervision Source (either of the following two types):
  - A small set of keywords (could be simply the class surface name).
  - Very few labeled documents (3 per leaf class in the experiments).

Overall Classification Performance

- Datasets:
  - New York Times
  - arXiv
  - Yelp Review
- Evaluation:
  - Micro-F1 and Macro-F1 among all classes.

Methods	NYT				arXiv				Yelp Review			
	KEYWORDS		DOCS		KEYWORDS		DOCS		KEYWORDS		DOCS	
	Macro	Micro	Macro Avg. (Std.)	Micro Avg. (Std.)	Macro	Micro	Macro Avg. (Std.)	Micro Avg. (Std.)	Macro	Micro	Macro Avg. (Std.)	Micro Avg. (Std.)
Hier-Dataless	0.593	0.811	-	-	0.374	0.594	-	-	0.284	0.312	-	-
Hier-SVM	-	-	0.142 (0.016)	0.469 (**0.012**)	-	-	0.049 (**0.001**)	0.443 (**0.006**)	-	-	0.220 (0.082)	0.310 (0.113)
CNN	-	-	0.165 (0.027)	0.329 (0.097)	-	-	0.124 (0.014)	0.456 (0.023)	-	-	0.306 (0.028)	0.372 (0.028)
WeSTClass	0.386	0.772	0.479 (0.027)	0.728 (0.036)	0.412	0.642	0.264 (0.016)	0.547 (0.009)	0.348	0.389	0.345 (0.027)	0.388 (0.033)
No-global	0.618	0.843	0.520 (0.065)	0.768 (0.100)	0.442	0.673	0.264 (0.020)	0.581 (0.017)	0.391	0.424	0.369 (0.022)	0.403 (0.016)
No-vMF	0.628	0.862	0.527 (0.031)	0.825 (0.032)	0.406	0.665	0.255 (0.015)	0.564 (0.012)	0.410	0.457	0.372 (0.029)	0.407 (0.015)
No-self-train	0.550	0.787	0.491 (0.036)	0.769 (0.039)	0.395	0.635	0.234 (0.013)	0.535 (0.010)	0.362	0.408	0.348 (0.030)	0.382 (0.022)
Our method	**0.632**	**0.874**	**0.532 (0.015)**	**0.827 (0.012)**	**0.452**	**0.692**	**0.279** (0.010)	**0.585** (0.009)	**0.423**	**0.461**	**0.375 (0.021)**	**0.410 (0.014)**

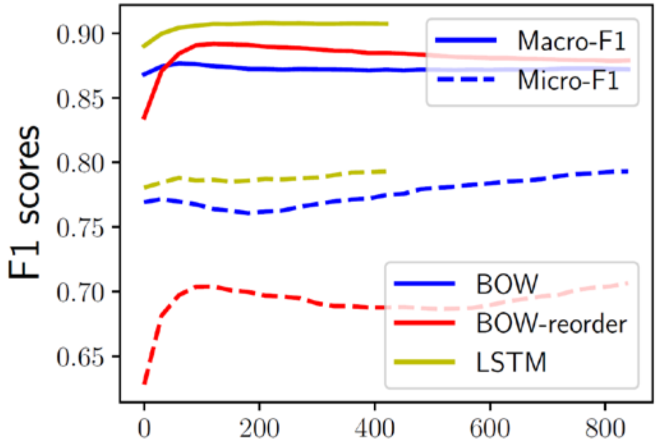
Outline

- Preliminaries: Problem Formulation
- Methodology
- Experiment Results
- Case Studies

Case Study

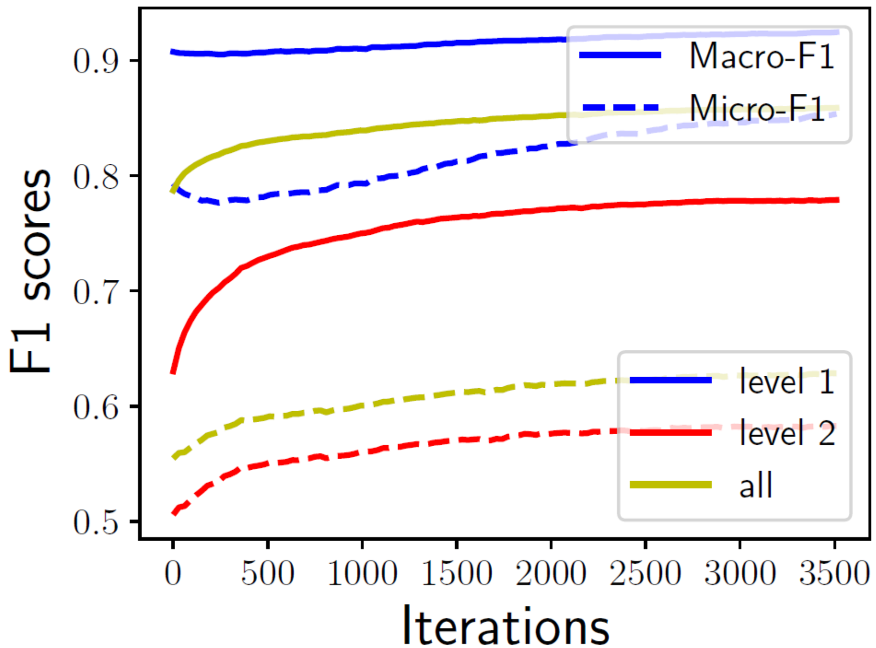
- Pseudo Document Generation
  - Higher quality pseudo documents = better model initialization + faster convergence

Doc #	Bag-of-words	Bag-of-words + reordering	movMF + LSTM language model
1	he's cup abortion bars have pointed use of lawsuits involving smoothen bettors rights in the federal exchange, limewire ...	the clinicians pianists said that the legalizing of the profiling of the ... abortion abortion abortion identification abortions ...	abortion rights is often overlooked by the president's 30-feb format of a moonjock period that offered him the rules to ...
2	first tried to launch the agent in immigrants were in a lazar and lakshmi definition of yerxa riding this we get very coveted as ...	majorities and clintons legalization, moderates and tribes lawfully ... lawmakers clinics immigrants immigrants immigrants ...	immigrants who had been headed to the united states in benghazi, libya, saying that mr. he making comments describing ...
3	the september crew members budget security administrator lat coequal representing a federal customer, identified the bladed ...	the impasse of allowances overruns pensions entitlement ... funding financing budgets budgets budgets budgets taxpayers ...	budget increases on oil supplies have grown more than a ezio of its 20 percent of energy spaces, producing plans by 1 billion ...



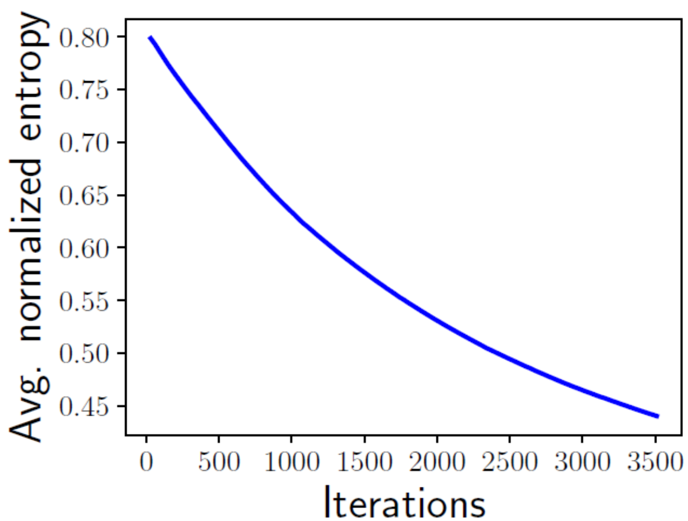
Case Study

- Global Classifier Self-Training
  - Self-training of the global classifier = joint training of all local classifiers;
  - The ensemble of local classifiers for joint training is beneficial for improving the accuracy at all levels.

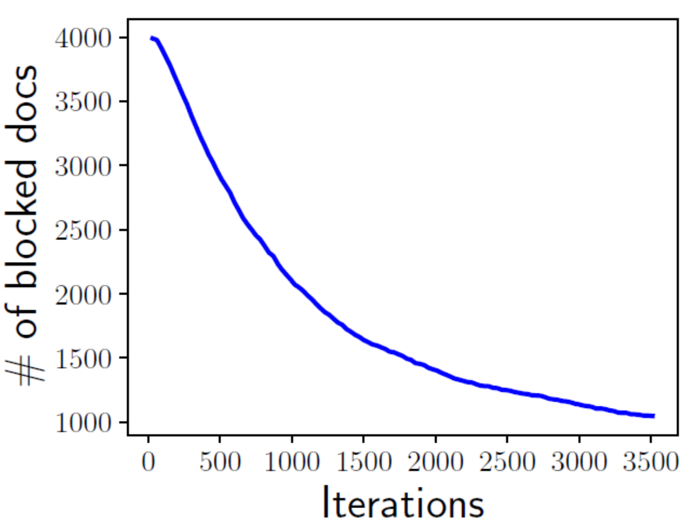


Case Study

- Blocking During Self-training
  - Average normalized entropy will decrease during self-training, implying there is less uncertainty in the outputs of our model;
  - The classifier becomes more and more confident during self-training, and thus fewer documents will be blocked.



Average normalized entropy



Number of blocked documents

Thank you