Zero-shot Text Classification via Reinforced Self-training

Zhiquan Ye, Yuxia Geng, Jiaoyan Chen, Xiaoxiao Xu, Suhang Zheng, Feng Wang, Jingmin Chen, Jun Zhang, Huajun Chen

ACL 2020

Motivation

- Zeroshot method
  - Embedding text and label into joint space
  - Matching text and label representation
- A self-training based method to leverage unlabeled data in zero-shot text classification
- A reinforcement learning framework to learn data selection policy automatically instead of using manually designed heuristics

Supervised Learning vs Zeroshot Learning

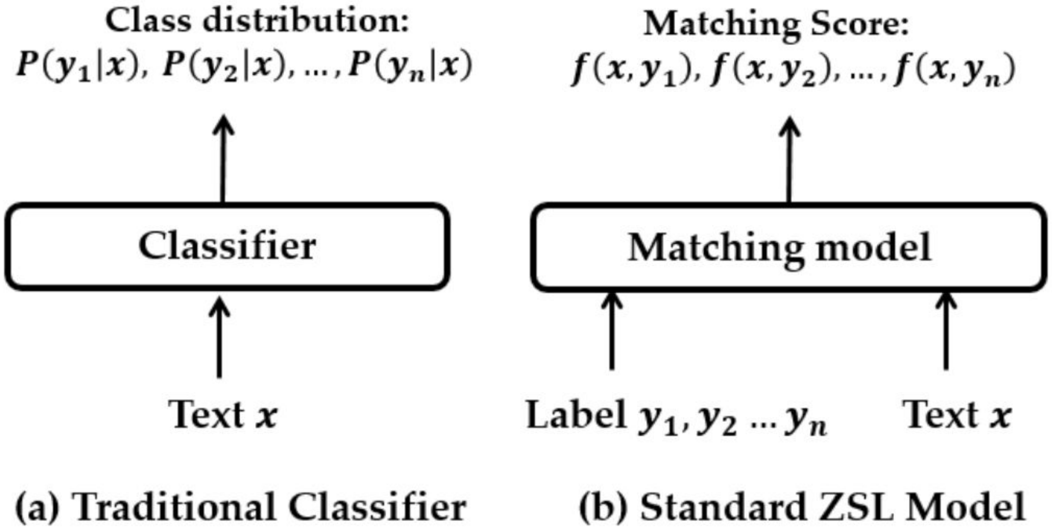

Figure 1: Illustration of the traditional classifier and standard ZSL model.

Model overview

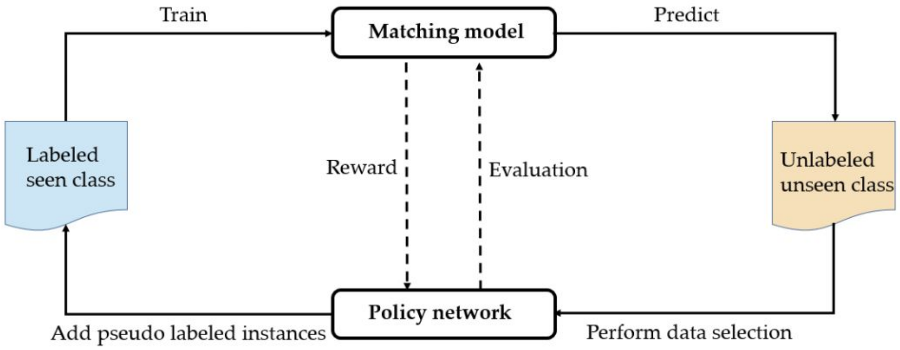

Figure 2: Overview of our reinforced self-training framework for zero-shot text classification.

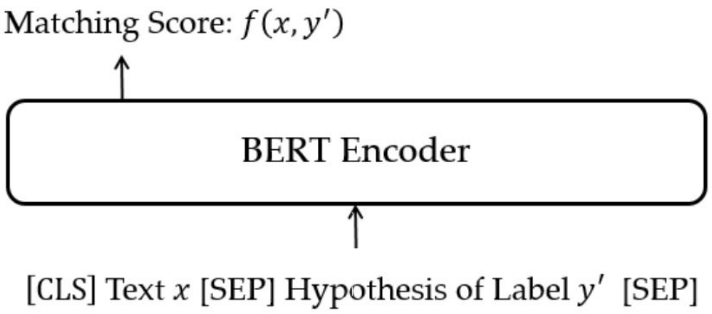

Figure 3: BERT as the base matching model.

Reinforcement Learning for Self-training

- Self-training:
  - Predict label on unlabeled data
  - Select samples with high confidence
- States
  - Prediction confidence p_{x,y^*};
  - Representation of text c_{x,y^*}
- Action
  - Select instance or not $P(a|s_t)$.
- Reward
  - Train the model on selected data, evaluate on dev set
  - Dev set contains labeled and unlabeled data

$$r_k = \frac{(F_k^s - \mu^s)}{\sigma^s} + \lambda \cdot \frac{(F_k^u - \mu^u)}{\sigma^u}$$

Reinforcement Learning for Self-training (2)

- Policy network

$$z_t = ReLU(W_1^T c_{x,y^*} + W_2^T p_{x,y^*} + b_1), \quad (5)$$

$$P(a|s_t) = softmax(W_3^T z_t + b_2)\ . \quad (6)$$

- Optimization

$$J(\phi) = E_{P_\phi(a|s)}[R(s,a)]\ ,$$

Algorithm 1 Reinforced self-training for zero-shot text classification

Require: labeled seen data $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^N$, unlabeled unseen data $\mathcal{D}^u = \{(x_i^u)\}_{i=1}^M$, seen validation set $\mathcal{D}_{dev}^s$.

Initialize pseudo-labeled data $\mathcal{D}^p \leftarrow \emptyset$
for $i = 1 \to N_1$ **do** //iteration i
  Train matching model f with instances from $\mathcal{D}^s$ and $\mathcal{D}^p$.
  Make prediction on $\mathcal{D}^u$, get confidence P.
  Get a subset Ω from $\mathcal{D}^u$ by ranked confidence P.
  for $j = 1 \to N_2$ **do** //episode j
    if early stop criteria is met **then**
      break
    end if
    Shuffle $\Omega = \{B_1, B_2, ..., B_{N_3}\}$.
    for $k = 1 \to N_3$ **do** //batch k
      Get a batch B_k from Ω.
      Decide action for each instance in B_k, get selected instances B_k^p.
      Train model f' with B_k^p.
      Evaluate on $\mathcal{D}_{dev}^s$ and $\mathcal{D}_{dev}^u$, get F_k^s, F_k^u.
    end for
    Compute rewards $\{r_k\}_{k=1}^{N_3}$ by equation 4.
    // update policy network
    for $k = 1 \to N_3$ **do**
      $\phi \leftarrow \phi + \eta \frac{r_k}{|B_k|} \sum_{t=1}^{|B_k|} \nabla_\phi logP(a_t|s_t)$
    end for
  end for
  $\mathcal{D}_i^p \leftarrow \cup_{k=1}^{N_3} B_k^p$
  $\mathcal{D}^p \leftarrow \mathcal{D}^p \cup \mathcal{D}_i^p$
  $\mathcal{D}^u \leftarrow \mathcal{D}^u \setminus \mathcal{D}_i^p$
  $\mathcal{D}_{dev}^u \leftarrow \mathcal{D}^p$.
end for

Results

	Topic		Emotion		Situation		E-commerce	
	I	II	I	II	I	II	I	II
Word2vec	35.50	35.33	4.77	11.45	40.67	36.33	53.09	55.47
Label similarity	34.62	36.14	10.63	16.89	54.56	37.45	59.04	55.89
FC	19.45	22.46	27.36	8.31	24.33	25.01	26.40	22.45
RNN+FC	9.68	13.41	15.45	3.15	15.58	14.09	25.76	18.15
BERT	57.07	45.50	16.86	10.21	60.23	34.15	58.05	66.47
BERT+self-training	72.21	62.90	31.96	**19.72**	69.00	49.30	65.14	76.72
BERT+RL	**73.41**	**65.53**	**36.98**	19.38	**73.14**	**52.44**	**70.63**	**80.32**