NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

School of Computer Science and Engineering
College of Engineering

TFIDF meets Deep Document Representation: A Re-Visit of Co-Training for Text Classification

Student: Chen Zhiwei **Supervisor:** Assoc Prof Sun Aixin

1. Objective

As of today, both TFIDF and word embedding based representations are sufficient to represent a document and they are from different feature spaces. This motivates us to re-visit co-training algorithm, to explore whether the two sets of document representations facilitate us to utilize unlabeled documents in text classification. As illustrated in Table 1, experiments were conducted to evaluate the effectiveness of co-training with different combinations of document representations and classifiers on two benchmark datasets (20 Newsgroup and Ohsumed).

Table 1: Combinations of Document Representations and Classifiers

Representation	Classifier
TFIDF, Doc2Vec, USE, BERT_p, ELMo_p	SVM, MLP, RF, XGB
BERT_s, ELMo_s	CNN

2. Methodology

The original co-training algorithm is designed for binary classification task, whereas 20 Newsgroup and Ohsumed are multi-class classification tasks. Thus, slight modifications are made as described in Algorithm 1.

The details of the modified co-training setting is as follow:

- Randomly sample 10% of training documents from each class to form L.
- The remaining 90% forms U, and U' is sampled from U with size 400.
- After each iteration, top p most confident documents from U' are added to L, where p=1 for first 10 iterations, and increase by 1 after every 10 iterations.
- k=40 iterations, so that after 40 iterations, 100 documents are added from U to L

4. Observations

Co-training with different representations shows very different results on 20 Newsgroup and Ohsumed

- Typical text classification task 20 Newsgroup:
  - 5 out of 10 combinations experimented showed improvement
  - The h1 and h2 of those improved combinations are often those that are performing well even when only trained normally (i.e. not in co-training setting)
- Challenging text classification task Ohsumed:
  - Benefit of co-training is very limited
  - Probably due to weak performance of h1 and h2, which hurts the quality of new documents added to L
  - With relatively high error rate in the predicted labels, the classifiers in the later iterations learnt from noisy labeled data

Algorithm 1: Modified Co-training Algorithm

```
Given:
    • a set $L$ of labeled training examples
    • a set $u$ of unlabeled training examples
Create a pool $U'$ of examples by choosing $u$ examples at random from $u$
 Loop for $k$ iterations:
    • Use $L$ to train a classifier $h_1$ that considers the $x_1$ portion of x
    • Use $L$ to train a classifier $h_2$ that considers the $x_2$ portion of x
    • Allow $h_1$ to make predictions on $U'$, and label top $p$ examples from $U'$
      according to the confidence score
    • Randomly choose $p$ examples from $u$ to replenish $U'$
    • Allow $h_2$ to make predictions on $U'$, and label top $p$ examples from $U'$
      according to the confidence score
    • Randomly choose $p$ examples from $u$ to replenish $U'$
    • Add these self-labeled examples to $L$
Define classifier $h_3$ by multiplying outputs of $h_1$ and $h_2$
```

3. Results of Co-training

	h_1	ER_{h1}	h_2	ER_{h2}	Pre	Rec	F_1	$\Delta F1_L$	ΔF_1	Acc	ΔAcc_L	ΔAcc
20Newsgroup	TFIDF+SVM	0.05	Doc2Vec+MLP	0.09	0.60	0.56	0.55	-0.13	-0.22	0.57	-0.12	-0.21
		0.05	USE+SVM	0.01	0.73	0.72	**0.72***	+0.04	-0.05	**0.74**	+0.05	-0.04
		0.03	BERT_s+CNN	0.02	0.72	0.72	**0.72***	+0.04	-0.08	**0.73**	+0.04	-0.08
		0.07	ELMo_s+CNN	0.02	0.66	0.65	0.65	-0.03	-0.12	0.66	-0.03	-0.12
	Doc2Vec+MLP	0.07	USE+SVM	0.00	0.59	0.57	0.56	-0.12	-0.13	0.58	-0.10	-0.13
		0.11	BERT_s+CNN	0.01	0.71	0.69	**0.69***	+0.01	-0.11	**0.71**	+0.03	-0.10
		0.15	ELMo_s+CNN	0.01	0.63	0.61	0.60	-0.06	-0.16	0.62	-0.06	-0.14
	USE+SVM	0.00	BERT_s+CNN	0.02	0.73	0.72	**0.71***	+0.04	-0.09	**0.73**	+0.06	-0.08
		0.00	ELMo_s+CNN	0.01	0.67	0.66	0.65	-0.03	-0.11	0.67	-0.01	-0.09
	BERT_s+CNN	0.03	ELMo_s+CNN	0.03	0.72	0.71	**0.71***	+0.03	-0.09	**0.72**	+0.04	-0.09
Ohsumed	TFIDF+SVM	0.12	Doc2Vec+MLP	0.33	0.27	0.23	0.21	-0.07	-0.38	0.44	-0.03	-0.23
		0.13	USE+SVM	0.26	0.39	0.22	0.21	-0.07	-0.38	0.42	-0.05	-0.25
		0.11	BERT_p+SVM	0.19	0.45	0.36	**0.37**	+0.09	-0.22	**0.52**	+0.03	-0.15
		0.10	ELMo_s+CNN	0.25	0.20	0.19	0.17	-0.11	-0.11	0.41	-0.06	-0.26
	Doc2Vec+MLP	0.24	USE+SVM	0.23	0.24	0.21	0.20	0.00	-0.18	0.42	+0.02	-0.12
		0.19	BERT_p+SVM	0.22	0.37	0.28	0.27	-0.08	-0.25	0.48	-0.01	-0.13
		0.29	ELMo_s+CNN	0.20	0.22	0.20	0.19	-0.01	-0.38	0.42	+0.02	-0.15
	USE+SVM	0.26	BERT_p+SVM	0.18	0.42	0.31	0.32	-0.03	-0.20	0.49	0.00	-0.12
		0.26	ELMo_s+CNN	0.22	0.25	0.21	0.18	-0.02	-0.39	0.41	+0.03	-0.16
	BERT_p+SVM	0.31	ELMo_s+CNN	0.32	0.24	0.24	0.22	-0.13	-0.30	0.44	-0.05	-0.17

Table 2: Co-training results of combined classifier h3. Results are in bold if the performance of h3 is better than either h1 or h2 in the evaluated combination. Paired t-test is conducted on F1 scores only (not on Acc) and * denotes statistically significant.