Taming Pretrained Transformers for Extreme Multi-label Text Classification

Chang et al (2020)

NLP reading club 2021/07/08

Key Points

- Xtreme text classification: given an input text instance, return the most relevant labels from an enormous label collection, where the number of labels could be in the millions or more
- Proposes the X-Transformer, which uses a 2 step process to reduce the label space
- Enables Deep Transformer models to work on a real life scenario product2query on Amazon, as well as giving SOTA results on benchmark datasets

Dataset info

Dataset	n_{trn}	n_{tst}	$\lvert D_{\text{trn}}\rvert$	$\lvert D_{\text{trn}}\rvert$	L	$\bar{L}$	$\bar{n}$	K
Eurlex-4K	15,449	3,865	19,166,707	4,741,799	3,956	5.30	20.79	64
Wiki10-31K	14,146	6,616	29,603,208	13,513,133	30,938	18.64	8.52	512
AmazonCat-13K	1,186,239	306,782	250,940,894	64,755,034	13,330	5.04	448.57	256
Wiki-500K	1,779,881	769,421	1,463,197,965	632,463,513	501,070	4.75	16.86	8192

Table 2: Data Statistics. n_{trn}, n_{tst} refer to the number of instances in the training and test sets, respectively. $\lvert D_{\text{trn}}\rvert, \lvert D_{\text{tst}}\rvert$ refer to the number of word tokens in the training and test corpus, respectively. L is the number of labels, $\bar{L}$ the average number of labels per instance, $\bar{n}$ the average number of instances per label, and K is the number of clusters. The four benchmark datasets are the same as AttentionXML [32] for fair comparison.

Modelling steps

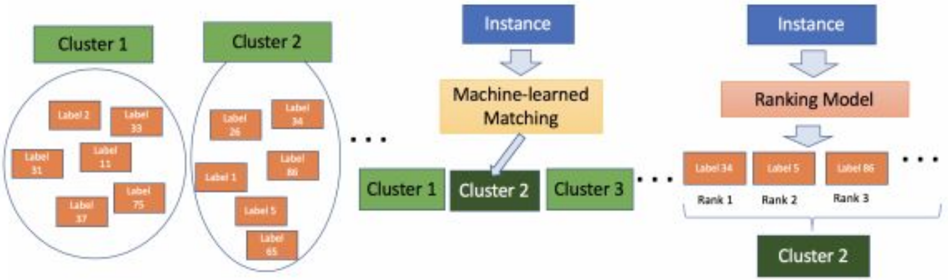

Figure 3: The proposed X-Transformer framework. First, Semantic Label Indexing reduces the large output space. Transformers are then fine-tuned on the XMC sub-problem that maps instances to label clusters. Finally, linear rankers are trained conditionally on the clusters and Transformer's output in order to re-rank the labels within the predicted clusters.

1. XLNet used to embed labels (using label text or label text + info from +ve instances), these are then clustered
2. A Transformer trained to predict cluster from training instances (num clusters <<< num labels)
3. Another transformer model is trained to rank the labels on each cluster (more robust training since you only need the samples that belong to each instance)

Experimental results

Eurlex-4K					Wiki-500K				
Method	Source	Relative Improvement over Parabel (%)			Method	Source	Relative Improvement over Parabel (%)		
		Prec@1	Prec@3	Prec@5			Prec@1	Prec@3	Prec@5
X-Transformer	Table 3	**+6.27%**	**+9.08%**	**+8.55%**	X-Transformer	Table 3	**+12.49%**	**+15.94%**	**+17.26%**
SLICE	[7, Table 2]	+4.27%	+3.34%	+3.11%	SLICE	[7, Table 2]	+5.53%	+7.02%	+7.56%
GLaS	[6, Table 3]	-5.18%	-5.48%	-5.34%	GLaS	[6, Table 3]	+4.77%	+3.37%	+4.27%
ProXML	[2, Table 5]	+3.86%	+2.90%	+2.43%	ProXML	[2, Table 5]	+2.22%	+0.82%	+ 2.92%
PPD-Sparse	[20, Table 2]	+1.92%	+2.93%	+2.92%	PPD-Sparse	[20, Table 2]	+2.39%	+2.33%	+ 2.88%
SLEEC	[9, Table 2]	-3.53%	-6.40%	-9.04%	SLEEC	[9, Table 2]	-29.84%	-40.73%	-45.08%

Table 6: Comparison of Relative Improvement over Parabel**. The relative improvement for each state-of-the-art (SOTA) method is computed based on the metrics reported from its original paper as denoted in the Source column.**

Resources

- [Github repo]
- [paper]