Paper Summary
Title:Exploiting Cloze Questions for Few Shot Text Classification and Natural Language Inference. arXiv:2001.07676v2
Authors:Timo Schick, Hinrich Schutze
Published: Jan 2020
Field:NLP

The authors use Pattern Exploiting Training(PET), a semi supervised training method to achieve state of the art performance

Three steps of PET:

1) Fine tune a language model on a small training set for each pattern
2) The assemble of all pattern trained model is used to label a large unlabeled dataset
3) A standard classifier is trained on the large labeled dataset.

A pattern *P* is a function that maps a sentence or set or sentences to a sentence with a single mask. A verbalizer *v* assigns any label to a word in the vocabulary. (*P,v*) pair (PVP), assigns words in masked location based on sentiment.

Eg: X=(Mia likes pie, Mia hates pie)
P(X)=Mia likes pie?, ___, Mia hates pie.
For which the task is to determine what word is appropriate in the masked location (yes,no).

PVP training:
Goal: Predict a label *l* given input *x*.
Use a masked language model *M* to assign a score to a word *w=v(l)*, given the Pattern *P*'s output (masked sentences). This is trained on a labeled training set *T*.
The score for a label is
M(v(l)| P(x)). The score and corresponding probability distribution over all labels is calculated as a softmax. The loss function is the cross-entropy. In practice a combined loss function is used:
$L = (1-a)\ L_{CrossEntropy} + a\ L_{maskedLanguageModel}$. With $a = 10^{-4}$ being the value used by authors.

Appropriate patterns are chosen for the task. A model is trained for each one. The ensemble (weight or uniform average) is used to label the large dataset.

An iterative approach can be used to improve performance *iPVP*.
$T^0 > M^0 > T^1 > M^1$ etc…, where the Pattern trained models M^0 are used to generate a new Training set T^1 from the large unsupervised dataset, which are used to generate a new set of Pattern trained models M^1 and so on.

Automated Verbalizer search

Given a verbalizer *v*, a token *t* is a good verbalization of label *l* if the probability that the model assigns *t* when and only when the label is *l*. A score is defined based on this probability. Then the best token is the one that maximizes the score.

1) Assign random verbalization to all labels multiple times (we do not want verbalizer to depend on initialization)
2) Repeatedly recompute best verbalization for all labels

Experiments

4 english datasets: Yelp reviews, AG's News, Yahoo Questions, MNLI.
Other languages: X-stance

RoBERTa as a language model for english, XLM-R for X-stance.
The datasets are made into training sets *T* of various sizes.
The labels are removed to create the unsupervised dataset *D*.
Comparing Supervised training vs PET:
Supervised training:
Learning Rate: 1E-5, batch size 16: sequence length: 256, for 250 steps.
PET:
Each batch is further divided into 4 label examples from training set *T* and 12 from unsupervised set *D*, to compute mixed loss function. The number of training steps is increased by 4.

Each dataset has 2~10 patterns. 1~2 verbalizers.
EG. Yelp Reviews is assigned 4 patterns
P1(a) = It was ___. a
P2(a) = a. All in all, it was ___.
P3(a)= Just ___! || a
P4(a) || In summary, the restaurant is ___.
With 1 verbalizer:
v(1)=terrible, v(2)=bad… v(5)=great.

Overall PET outperforms RoBERTa large and supervised models at all sizes (0-1000) of *T*. The difference between supervised and PET and iPET begins marginal at high training set sizes. AVS decreases PET performance when compared to hand picked verbalizers.

According to https://arxiv.org/abs/2009.07118 by same authors, ALBERT with PET/iPET (223M param) outperforms 76.8 vs 73.2 GPT-3(175000M param) on SuperGLUE (reading comprehension natural language understanding). For this paper, the PET is generalized to accommodate multiple masks.