Semi-Supervised Learning and Domain Adaptation in Natural Language Processing

6.3 META-ANALYSIS
6.4 PERFORMANCE AND DATA CHARACTERISTICS
6.5 DOWN-STREAM EVALUATION

吉田拓夢

6.3 META-ANALYSIS

It is applicable when you assume the individual studies estimate the same effect size

if we want to estimate an effect from a large set of studies, the average effect across all the studies will put too much weight on results obtained in small or low-quality experiments in which you typically see more variance

meta-analysis
- fixed-effect model
- random-effect model

It stems from the observation that...

If you cannot make that assumption because the studies may differ in various aspects, leading the within-study estimates to be estimates of slightly different effect sizes

6.3 META-ANALYSIS

In the fixed effects model we weight the effect sizes T1,...,Tm (or accuracies, in case) by something related to sample size.

the combined effect size T

$$\hat{T} = \frac{\sum_{i \geq 1}^{M} \omega_i T_i}{\sum_{i \geq 1}^{M} \omega_i}$$

the variance of the combined effect

$$v = \frac{1}{\sum_{i \geq 1}^{M} \omega_i}$$

They weight each of their results by the inverse of the variance vi in the study

the 95% confidence interval

$$\hat{T} \pm 1.96 \sqrt{v}$$

6.3 META-ANALYSIS

In the random effects model we replace the variance vi with the variance plus between-studies variance τ^2:

with degree of freedom df=N-1

$$\tau^2 = \frac{\sum_{i\geq 1}^{k} \omega_i T_i^2 - \frac{\left(\sum_{i\geq 1}^{k} \omega_i T_i\right)^2}{\sum_{i\geq 1}^{k} \omega_i} - df}{\sum_{i\geq 1}^{k} \omega_i - \frac{\sum_{i\geq 1}^{k} \omega_i^2}{\sum_{i\geq 1}^{k} \omega_i}}$$

all negative values are replaced by 0

6.3 META-ANALYSIS

Let us try to compute the 95% confidence interval for a series of 25% randomly extracted cross-domain problem instances 20 Newsgroups.

We can use meta-analysis to estimate effect sizes.

The code is presented in Figure 6.1.

In a fixed effects model, the 95% confidence interval estimated on 25 randomly extracted problem instance that Naive Bayes is better than perceptron is [3.9%, 5.2%].

The weighted mean is 4.6% and the macro-average is 3.9%.

Using a random effects model on the same 25 datasets,

the 95% confidence interval becomes [-6.5%, 6.6%].

Figure 6.1

```
_r1, _r2, _r3=[], [], []

for _ in range(25):
  X_train, y_train, X_test, y_test=TwentyNewsgroups()

  clf=perc()
  clf.fit(X_train, y_train)
  y_hat=clf.predict(X_test)
  r1=metrics.zero_one_score(y_test, y_hat)
  _var=[]
  for _ in range(50):
    sample=np.random.randist(0, y_test.shape[0], size=y_test.shape[0])
    _var.append(metrics.zero_one_score(y_test[sample], y_hat[sample]))
  w_i=1/np.arry(_var).var()
  _r1.append(r1)
  _r2.append(w_i*r1)
  _r3.append(w_i)

T=sum(_r2)/sum(_r3)

thau_squared=(sum(_r2_squared)-((sum(_r2)**2)/sum(_r3))-24)/¥
(sum(_r3)-(sum(_r3_squared)/(sum(_r3))))
if thau_squared < 0:
  thau_squared=0
_r4=[]
for old_w_i in _r3:
  _r4.append(1/old_w_i+thau_squared)
v=1/sum(_r4)

print "macro-av.:", sum(_r1)/len(_r1)
print "weightd mean:", T
print "95% conf.int.:", T-1.96*sqrt(v) "<-->", T+1.96*sqrt(v)
```

6.3 META-ANALYSIS

Using macro-average and meta-analysis to predict error reductions on document classification datasets based on k observations.

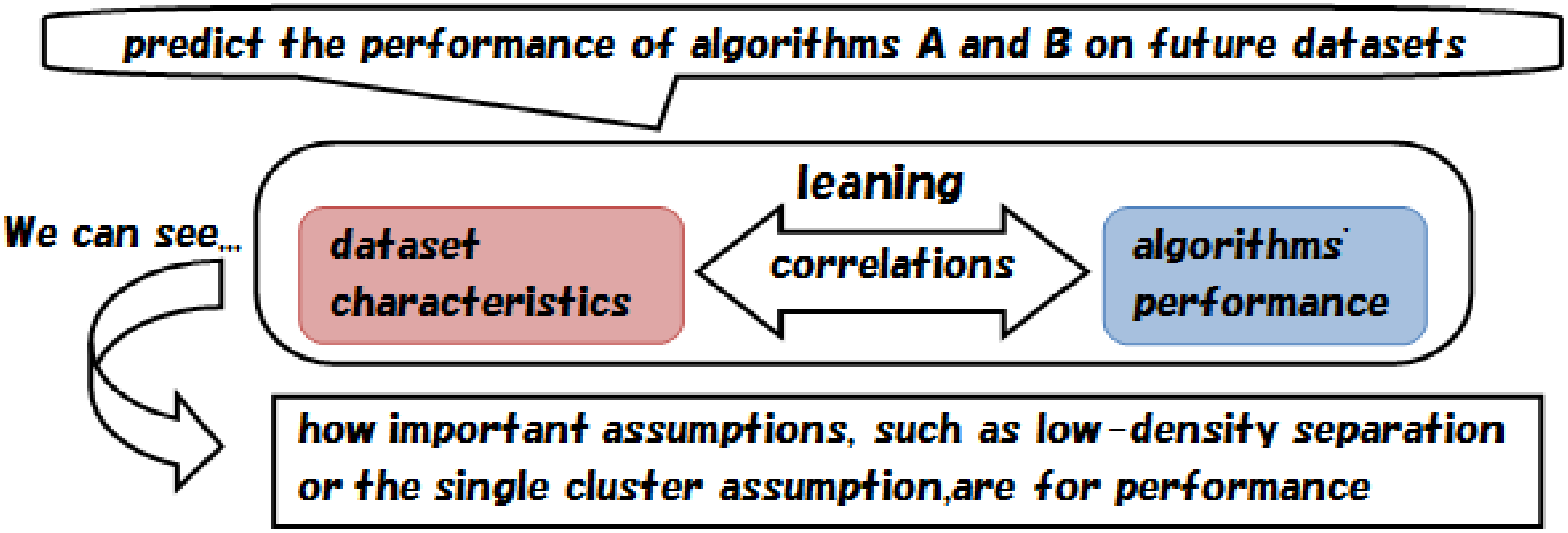
	macro-av	fixed	random
$k = 5$			
err.	-0.1656	-0.0350	-0.0428
p-value	-	< 0.001	< 0.001
$k = 10$			
err.	-0.1402	-0.0329	-0.041
p-value	-	< 0.001	< 0.001
$k = 15$			
err.	-0.0809	-0.0799	-0.0804
p-value	-	< 0.001	< 0.001

meta-analysis providers much better estimates than macro-averages across the board

In each experiment, they randomly select k datasets and estimate the true effect size using macro-average, a fixed effects model, a random effects model, and a corrected random effects model

They evaluate their estimates against the observed average effect across five new randomly extracted datasets.
For each k, they repeat the experiment 20 times and report average error.

6.4 PERFORMANCE AND DATA CHARACTERISTICS



	correlation coefficients		95% confidence interval
	macro-av	weighted mean	
their first run	-0.088	-0.124	[-0.131 , -0.117]
perceptron	-0.075	-0.0739	[-0.082 , -0.066]

over 10 randomly extracted datasets

6.5 DOWN-STREAM EVALUATION