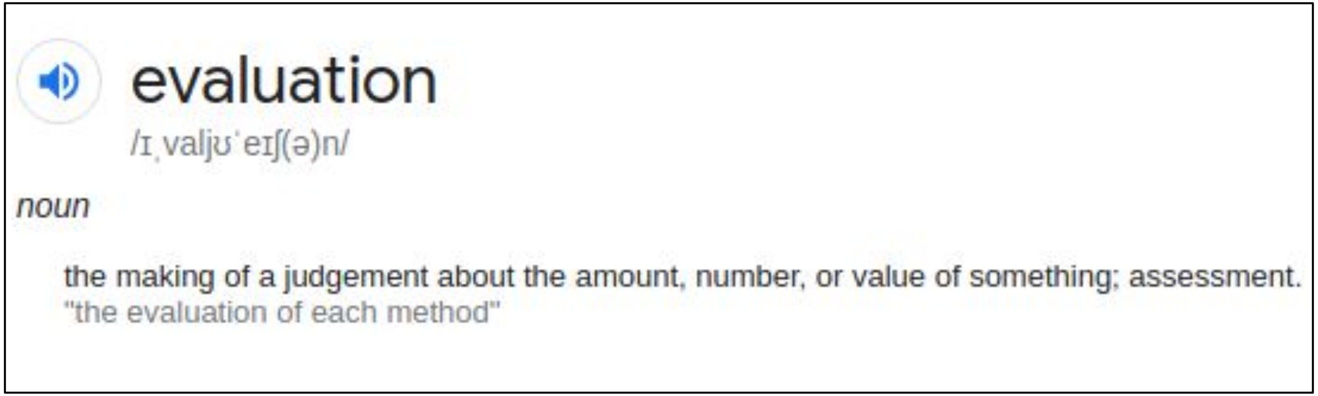
Evaluation: Performance Evaluation Part 1

COMP61332: Text Mining
Week 5
Riza Batista-Navarro

What does "evaluation" mean?



evaluation

/ɪˌvaljʊˈeɪʃ(ə)n/

noun

the making of a judgement about the amount, number, or value of something; assessment.
"the evaluation of each method"

In Computer Science, synonymous to: **testing**

Why evaluate?

Users: want to compare different available solutions

in the context of their needs/problems

want the best solution there is

Developers: want to know about progress/advancement

check for improvement

look for the better/best algorithm

Main types of evaluation

Performance evaluation

How well is a system doing against an ideal state (benchmark)?

Adequacy evaluation

Is the system fit for purpose? Does it do what the user wants (within cost, time)?

Diagnostic evaluation

Any side effects from recent updates?

Use of test suites

Performance Evaluation

kaggle

SEMEVAL

SemEval: Semantic Evaluations

Often based on a **benchmark data set**

Organised around a community challenge/shared task

- Specific task is **defined**
- **Gold standard** data provided: training, development, test

Automated means for scoring submissions, whereby the following are compared:

response: system-generated annotations/predictions

reference: gold standard

Gold Standard Data

Time-consuming and costly to produce

Requires annotation instructions (**annotation guidelines**)

Annotations done by experts

- May need some training in linguistics
- **Multiple annotators** need to label the same samples (a subset)

Need to ensure that annotations are **reliable**

- Calculate **inter-annotator agreement**
- There might be disagreements

Inter-annotator agreement (IAA): Kappa coefficient

Takes individual bias into consideration: annotators have subjective interpretations of annotation guidelines

$$\kappa = \frac{P(a) - P(e)}{1 - P(e)}$$

***P(a)* = observed agreement**, proportion of times judges agreed

***P(e)* = expected agreement**, proportion of times judges expected to agree by chance

Inter-annotator agreement (IAA): Kappa coefficient

Assume:

we have two annotators *A1* and *A2*

they are providing annotations for a binary classification task: does a sample belong to some class *c*? yes or no

$$\kappa = \frac{P(a) - P(e)}{1 - P(e)}$$

P(a) = P(A1=yes, A2=yes) + P(A1=no, A2=no)

P(e) = P(A1=yes)*P(A2=yes) + P(A1=no)*P(A2=no)

Kappa coefficient: Example

	Annotator 1		
	yes	no	total
			40

Kappa coefficient: Example

	Annotator 1		
	yes	no	total
	33	7	40

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes			
	no			
	total	33	7	40

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes	31		
	no	2		
	total	33	7	40

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes	31	1	
	no	2	6	
	total	33	7	40

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes	31	1	32
	no	2	6	8
	total	33	7	40

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes	31	1	32
	no	2	6	8
	total	33	7	40

P(a) = P(A1=yes, A2=yes) + P(A1=no, A2=no)

= (31/40) + (6/40)

= 0.925

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes	31	1	32
	no	2	6	8
	total	33	7	40

$$P(e) = P(A1=yes)*P(A2=yes) + P(A1=no)*P(A2=no)$$

$$= ((33/40) * (32/40)) + ((7/40) * (8/40))$$

$$= 0.695$$

Kappa coefficient: Example

		Annotator 1		
		yes	no	total
Annotator 2	yes	31	1	32
	no	2	6	8
	total	33	7	40

Kappa = (P(a)-P(e))/(1-P(e))

= (0.925-0.695)/(1-0.695) = 0.754

Kappa coefficient: Interpretation

Landis and Koch, 1977

slight < 0.2 < fair < 0.4 < moderate < 0.6 < substantial < 0.8 < perfect

Grove et al., 1981 (psychiatric community)

0.6 < acceptable

Krippendorff, 1980

0.67 < tentative conclusions < 0.8 < definite conclusions

Rietveld and van Hout, 1993

0.4 < moderate < 0.6 < substantial < 0.8

Green, 1997

low < 0.4 < fair/good < 0.75 < high

Other coefficients for IAA?

Scott’s Pi

P(e): different chance for different categories

Fleiss’ Kappa

multi-annotator generalisation of (Cohen's) Kappa and Scott’s Pi

These would work if we can define negative cases; for some tasks this is not possible, e.g., **NER**

For such tasks, **F-score** is reported instead