https://presemo.helsinki.fi/nlp2020

LECTURE 3: EVALUATION

Mark Granroth-Wilding

POS-TAGGING REMINDER

PTB tags

Return	now	to	your	quarters	and	I	will	send	you	word	of	the	outcome
V	Adv	Prep	Pro	N	CC	Pro	V	V	Pro	N	Prep	Det	N
VB	RB	IN	PP$	NNS	CC	PP	MD	VB	PP	NN	IN	DT	NN

- Word may take many POS tags
- Model tries to **disambiguate**
- Use statistics from *corpus*
- Many models: features, statistics, . . .

POS	**Example**
Noun	*The dog ate the bone*
Verb	*The dog ate the bone*
Adjective	*The big dog ate the bone*
Adverb	*He ate the bone quickly*
Pronoun	*He ate my bone*
Determiner	*The dog ate that bone*
Preposition	*He chewed with his teeth*
Coordinating conjunction	*He chewed and growled*

POS TAGGER OUTPUT

- You've got a new POS tagging method
- Trained model
- Try on some example inputs...
- Output looks good! – better than existing method

Questions:

- **Real** improvement, or did we get lucky?
- **How big** is improvment?
- **How general** is improvment?
  Will it help in practice on lots of real data?

EVALUATION

- Scientific method: make **hypotheses**, test them
- Evaluation = testing
- Why?
  - **Public review**: convince others your method is good
  - **Internal review**: unbiased validation of improvement
  - And **quantification**
  - Systems might **evaluate themselves** to improve

EVALUATION IN NLP

- Much of today probably familiar from e.g. ML courses
- We'll cover much from scratch
- Some aspect specific to NLP
- Some NLP issues require special attention

EVALUATION

(Close as possible)

- Unbiased system comparison
- Compare: *models*, *training sets*, *methods*, *frameworks*
- Use existing evaluation if possible:
  test data, methods, metrics
- Design evaluation before developing model: success criteria
- Desiderata:
  - **Unbiased**
  - **Reliable**: enough data
  - **Representative** of target domain
    (often: as general as possible)
  - **Reproducible**: others can repeat, compare other models

---

TYPES OF EVALUATION

Evaluating *method* for a *task*

Intrinsic: test inherent to task

- POS tagging: tag accuracy
- Parsing: dependency F-score (see later)
- Search: recall of relevant documents

Extrinsic: test of effect on other, *downstream* tasks

- POS tagging: affect on parsing performance
- Parsing: improvement in relations extract from text (IE)
- Search: perceived improvement of tool for users

REUSING TEST DATA

- Many tasks: standard test sets exist
- Advantages of reusing test data:
  - **Direct comparison** to earlier work
  - Laborious **annotation** work reused ← More later
  - Avoid **unintentional bias** ← E.g. towards your method
- Drawbacks: later today

STANDARD EVALUATION DATASETS

- Examples for specific tasks:
  - *Penn Treebank*: syntactic parsing
  - *Brown Corpus*: POS tagging (and others)
- Preparation and annotation: huge effort
  - E.g. PTB: millions of words with syntactic trees
  - Annotated over 3 years
- Often carefully chosen data: representative, balanced, etc

SHARED TASKS

- **Competition** on particular task
- Provided: ← Often available afterwards Reuse for later evaluation
  - Training data
  - Detailed task description
  - Test data (released later)
- Compare, publish best-performing methods
- E.g. SemEval-2018
  - *Irony detection in Tweets*
  - *Argument reasoning comprehension* (resources)
  - *Semantic relation extraction in scientific papers*

HOW DO WE KNOW WHAT'S GOOD?

Input sentence

POS tagger 1

POS tagger 2

Gold standard annotations

	He	*gazed*	*again*	*at*	*the*	*scene*	*below*	*,*	*and*	*now*
1:	PRP	VBD	RB	IN	DT	NN	JJ	,	CC	RB
2:	PRP	VBD	RB	IN	DT	NN	IN	,	CC	RB
GS:	PRP	VBD	RB	IN	DT	NN	JJ	,	CC	RB

noted	*one*	*difference*	*from*	*the*	*accustomed*	*Porovian*	*landscape*	*.*
VBN	CD	NN	IN	DT	JJ	NNP	NN	.
VBN	CD	NN	IN	DT	VBN	JJ	NN	.
VBN	CD	NN	IN	DT	JJ	JJ	NN	.

	Num correct	Total tags	**Accuracy**
System 1	16	18	16/18 = 89%
System 2	15	18	15/18 = 83%

METRIC: ACCURACY

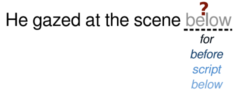
- Simple metric: **accuracy**

$$\frac{\#\text{ correct}}{\text{total words}}$$

- Measure over *large* test corpus
- Not always suitable
- Other suitable in many circumstances
- Some more specific to language tasks: coming up

PERPLEXITY

E.g. language model:
probability dist over missing word



- Sometimes model assigns *scores* to possible outcomes
  - Often *probabilities*
- *Observed* outcome should receive high probability
- but others are not *wrong*
- Models produce **distributions** over outcomes ← words
- Compare **probabilities** of **observed outcomes** ← words
- Idea: measure mean probability of observations ← observed words
- Typical measure: **perplexity**

PERPLEXITY

- Information theoretic measure
- How **surprised** model is on average by each observation

$$2^{-\frac{1}{m}log_2(p(w_0,...,w_{m-1}))}$$

- Lower is better
- Will see later: *language model* evaluation

METRICS

- *Accuracy* and *perplexity* are **evaluation metrics**
- Compare output to *gold standard*
- Numeric measure of correctness
- Different metrics appropriate for different tasks
- Sometimes multiple: capture
  - different aspects of task
  - different types of 'correctness'

SOME OTHER METRICS

- Other commonly used metrics:
  - **P**recision
  - **R**ecall
  - **F**-score
- Common generally in ML
- Very common in NLP evaluation

PRECISION & RECALL

- **Task**: detect relevant items in large corpus
- E.g. information retrieval / search ← More in lecture 9
- **Gold standard**: annotated set of all relevant
- System tags subset of items as *relevant*

True positives

$$\text{Precision (P)} = \frac{\text{\# relevant GS items tagged as relevant}}{\text{\# items tagged as relevant}}$$

$$\text{Recall (R)} = \frac{\text{\# relevant GS items tagged as relevant}}{\text{\# GS relevant items}}$$

PRECISION & RECALL

- **Task**: detect relevant items in large corpus
- E.g. information retrieval / search ← More in lecture 9
- **Gold standard**: annotated set of all relevant items
- System tags subset of items as *relevant*

$$\text{Precision (P)} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

$$\text{Recal (R)} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

PRECISION & RECALL

$$\text{Precision (P)} = \frac{\text{\# relevant GS items tagged as relevant}}{\text{\# items tagged as relevant}}$$

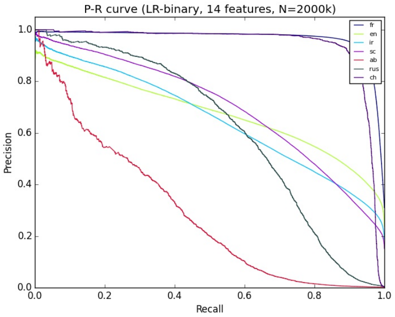
How much can we rely on returned results being correct?

$$\text{Recall (R)} = \frac{\text{\# relevant GS items tagged as relevant}}{\text{\# GS relevant items}}$$

How much can we rely on system to find all correct results?

- Sometimes one more important than other
- Often systems can trade off between them

PRECISION-RECALL CURVE



Graph from KubiK888 on SO

F-SCORE

- Combine into one metric: **F-score** / **F-measure**
- Harmonic mean of P and R

$$F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

- Most often, just use $\beta = 1$:

$$F_1 = \frac{2PR}{P + R}$$

EXERCISE

POS-tagger output

Discuss:

- What metric(s) is suitable to evaluate this output?
- Why?
- How well does this tagger do?

Word	Tagger's answer	Correct answer
“	PUNCT	PUNCT
It	NOUN	NOUN
's	PART	VERB
too	ADV	ADV
bad	ADV	ADJ
that	ADP	ADP
Myles	NOUN	NOUN
Cabot	NOUN	NOUN
can	VERB	VERB
't	ADV	ADV
see	ADJ	VERB
this	DET	DET
!	PUNCT	PUNCT
”	PUNCT	PUNCT
I	VERB	NOUN
exclaimed	NOUN	VERB
,	PUNCT	PUNCT
as	PREP	ADP
my	ADJ	ADJ
eye	VERB	NOUN
fell	NOUN	VERB
on	PREP	PREP
the	ADJ	DET
following	ADV	ADJ
item	NOUN	NOUN
:	PUNCT	PUNCT

EXERCISE

NER output

Discuss:

- Is a different metric suitable to evaluate this output?
- If so, why?
- How well does this tagger do?

Word	Tagger's answer	Correct answer
“	NONE	NONE
It	NONE	NONE
's	NONE	NONE
too	NONE	NONE
bad	NONE	NONE
that	NONE	NONE
Myles	BPERS	BPERS
Cabot	IPERS	IPERS
can	IPERS	NONE
't	NONE	NONE
see	NONE	NONE
this	NONE	NONE
!	NONE	NONE
”	NONE	NONE
I	NONE	NONE
exclaimed	NONE	NONE
,	NONE	NONE
as	NONE	NONE
my	NONE	NONE
eye	BLOC	NONE
fell	ILOC	NONE
on	NONE	NONE
the	NONE	NONE
following	NONE	NONE
item	NONE	NONE
:	NONE	NONE

BASELINES

- You've got a brand new, swanky POS tagger
- Output looks good, but how reliable is this?
- Evaluate on unseen test set
- Measure accuracy
- Compare to existing tagger: better!
- Looks good, but how hard is this task/dataset?

System	Accuracy (%)
MyTagger	94.5
OldTagger	92.3

BASELINES

- How hard is this task/dataset?
- Also compare to **baselines**
- Simple models using much less data, inference, training time. . .
- Tells us
  - how much these things are buying us
  - how hard task is
  - whether we're wasting our time

A SIMPLE BASELINE

Some baselines for POS tagger:

- Random choice
  - 20 tags → expect 5%
- Majority class
  - Most frequent tag: `Noun`
  - 61.5% of words in test set
  - Acc 61.5% if always `Noun`
- Ignore context
  - For each word, use its most frequent tag in training set
  - '*Unigram*' model (like unigram-HMM)

System	**Acc (%)**
MyTagger	94.5
OldTagger	92.3
Random	5
Majority class	61.5
Unigram	90.2

A SIMPLE BASELINE

- In this case, unigram does well
- Neither tagger does *much* better
- That 2-5% could be very hard
- Need some error analysis
- Maybe need
  - evaluation/analysis that explains differences
  - harder test set, focussed on difficult cases

System	**Acc (%)**
MyTagger	94.5
OldTagger	92.3
Random	5
Majority class	61.5
Unigram	90.2

CEILINGS

- Baslines establish *lower bounds*
- **Ceiling**: *upper bound*
- Best we can ever expect from model
- Probably unachievable goal
- Human performance: agreement between annotators
- Or perfect knowledge of some part of task
  - Quantify how hard parts of the task are
  - How well model does on different parts

SIGNIFICANCE

- Measuring things, comparing measurements
- What and how depends on task
- Important question in all cases:

  Are the differences ***significant***?
- Are the differences down to chance? Or something more?

Test data
Stochastic training
. . .

Is a model *significantly* better than *baseline*?
Is it *significantly* worse than *ceiling*?

Based on slides by Henry Thompson, Edinburgh University, with permission

---

EXAMPLE: TOSSING A COIN

- Toss a coin **40** times
- Get **heads 17** times, **tails 23**
- Expected value of fair coin: **20**
- Is different *significant*?
  - Yes → (probably) not a fair coin
  - No → (probably) a fair coin

Based on slides by Henry Thompson, Edinburgh University, with permission

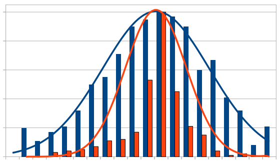
---

NORMAL DISTRIBUTIONS

- Significance measurement is a complex area
- Simple basic idea is simple:

  Measure difference in terms of standard deviations

- **Standard deviation**: how *representative* is mean?
- More outliers / further from mean → mean less representative
  → higher standard deviation

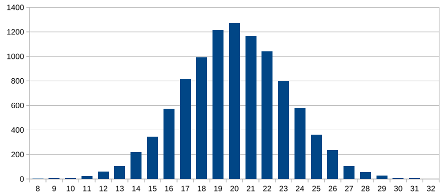


Based on slides by Henry Thompson, Edinburgh University, with permission

---

DISTRIBUTION OF A FAIR COIN

- Many coin toss trials: e.g. toss 40×, repeat 100k×
- Distribution approaches **normal distribution**
- Mean ≃ 50%, but range of outcomes plausible

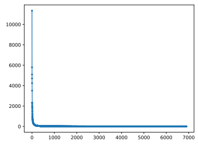


- Even result >2 std devs out will come up ∼1/20 trials

Based on slides by Henry Thompson, Edinburgh University, with permission

WHICH SIGNIFICANCE TEST?

- **Parametric** when underlying distribution is **normal**
  - **t-test**, **z-test**, . . .
  - Check test assumptions appropriate for specific case
- **Non-parametric** otherwise
  - **McNemar's test** or variants
  - Usually non-parametric in NLP: *Zipf's law*, not normal



Based on slides by Henry Thompson, Edinburgh University, with permission

---

GOLD STANDARD DATA

- Data annotated with 'true' labels
- Human judgement of correct labels, ratings, . . .
- GS data often need for training
  - Supervised methods
- Also for evaluation, as we've seen

Important: evaluation is always on **unseen** data

TEST DATA

- **Unseen test data** is sacrosanct
- Unseen
  - by *model*: separate from training data
  - by *you*: avoid influencing model/method design, feature selection, . . .
  - by *evaluation routine*: don't run during development, only in final comparison
- All violations can produce **overfitting** in some form
  - How?

TRAINING AND TESTING

- **Overfitting** → model good on test set, not on other data
- Overestimates performance on unseen data, 'in the wild'
- Separate **training** and **test** sets
- Typical procedure:
  1. Collect GS data (annotation)
  2. Split into training+test (e.g. 80 : 20 or 90 : 10)
  3. Training set: examine data, train models, compare, . . .
  4. Test set: evaluate and report results

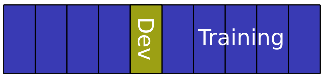
DEVELOPMENT

- Models have *hyperparameters*: want to set empirically

  E.g. *n* in n-gram LM, or smoothing parameters
- Need to test models before final evaluation
  - Compare model types, architectures, etc
  - Check modelling hypotheses
  - Test for overfitting
- Common solution: keep aside part of training set
- **Development** / **validation** set (devset): e.g. 80 : 10 : 10

LIMITED DATA

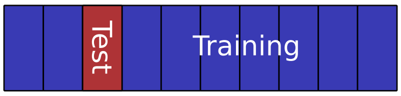
- Small corpus: splitting can leave too little for testing
- Or for training
- Alternative: **cross validation**



- Still need held-out **test set** for final evaluation

CROSS VALIDATION

- Can also use cross validation for **evaluation**
- Useful when annotated data extremely scarce
- More test data: results more representative
- Results less sensitive to random split
- But caution!
- Whole corpus must be **blind**: don't look at it during development
- Harder to avoid *bias* in long run

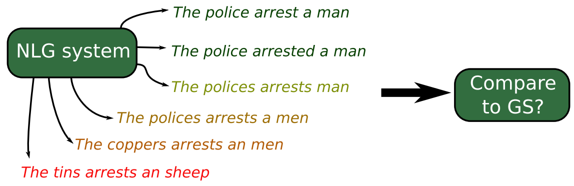


CHOOSING A DATASET

- Consider existing training/test sets
  - Someone else has done hard work for you
  - Usually carefully thought out: balanced, etc
- Things to consider:
  - **representative** of target domain
  - **annotated**/suitable to annotate
  - **large** enough
  - **balanced** (in various ways)
- If necessary, annotate: a lot of work

HUMAN EVALUATION

- Alternative: get people to asses system output
- Often replicating this with data-driven evaluation
  (Not always: sometimes data removes bias)
- Sometimes necessary or preferable



HUMAN EVALUATION

- Many difficult issues involved: no time today
- Some challenges:
  - Difficult to replicate/compare to later
  - Smaller test set: less representative
  - Costly
- **Inter-annotator agreement**
  - Important to measure: often low
  - Multiple judgements per example
  - Lower IAA: more judgements

CROWD SOURCING

- Popular for **human evaluation** and **annotation**
- Small money for small tasks
- E.g. *Amazon Mechanical Turk*
- *Much* less reliable than experts / known evaluators
- Cheap annotation of lots of data
- Many issues with quality
  - Need **redundancy**: lots of annotations per example
  - **Test questions** and other checks for bad/malicious annotators

---

ERROR ANALYSIS

- Broad metrics give overview, but don't tell us everything
- Beware *Zipf*: rare stuff often important
- **Error analysis**: more than just summary statistics
- More insight into what's going on
  - Find patterns in results:
    e.g. better scores on particular type of inputs
  - Qualitative insight: what type of mistakes does system make?
    → Different models capture different things?
  - Find bugs
  - Focus future work
- Evaluation sanity check: are 'better' models actually better?

CONFUSION MATRICES

Good way to find common problems

Example: POS tagging errors

Predicted labels

Correct labels

	IN	JJ	NN	NNP	RB	VBD	VBN
IN	–	2			7		
JJ	2	–	33	21	17	2	27
NN		87	–				2
NNP	2	33	41	–	2		
RB	22	20	5		–		
VBD		3	5			–	44
VBN		28				26	–

Diagonal: correct predictions

Example from J&M2 p190

CONFUSION MATRICES

Good way to find common problems

Example: POS tagging errors

Predicted labels

Correct labels

	IN	JJ	NN	NNP	RB	VBD	VBN
IN	–	2			7		
JJ	2	–	33	21	17	2	27
NN		87	–				2
NNP	2	33	41	–	2		
RB	22	20	5		–		
VBD		3	5			–	44
VBN		28				26	–

High values: common mistakes

Example from J&M2 p190

CONFUSION MATRICES

	IN	JJ	NN	NNP	RB	VBD	VBN
IN	–	2			7		
JJ	2	–	33	21	17	2	27
NN		87	–				2
NNP	2	33	41	–	2		
RB	22	20	5		–		
VBD		3	5			–	44
VBN		28				26	–

Finland

dog

Proper nouns (NNP) tagged as **common nouns** (NN).

Hard to distinguish in some contexts.
Important for **IE** and **MT**.

CONFUSION MATRICES

	IN	JJ	NN	NNP	RB	VBD	VBN
IN	–	2			7		
JJ	2	–	33	21	17	2	27
NN		87	–				2
NNP	2	33	41	–	2		
RB	22	20	5		–		
VBD		3	5			–	44
VBN		28				26	–

eaten

ate

Past participles (VBN) confused with **past tense** (VBD).

Important for finding edges of noun phrases:
Estimated counts are better than no counts.
He estimated counts for all categories.

THIS WEEK'S ASSIGNMENT

- Implement larger NLU pipeline
- Builds on last week's use of tools
- Evaluation against labelled data

- Help session: **Thursday, 9-11**
- Due **next Monday**

SUMMARY

- Evaluation important for:
  - Public review
  - Internal review, quantification of improvements
- Unbiased comparison of *models*, *data*, *methods*
- **Intrinsic** vs **extrinsic**
- **Shared tasks** and common test sets

SUMMARY

- Metrics:
  accuracy, *perplexity*, *P*, *R*, *F-score*
- **Baselines** & **ceilings**
- **Significance testing** in NLP
- Splitting test data: *training*, *test*, *development*, *cross-val*
- **Human evaluation**
- **Error analysis**: confusion matrices

Continue in next lecture with *annotation*

READING MATERIAL

- Evaluation intro, P, R, F-score: *J&M3 4.7*
- Intrinsic/extrinsic eval, data split, perplexity: *J&M3 3.2*
- Examples of shared tasks:
  http://www.conll.org/2019-shared-task
- Significance testing:
  Hitchhiker's Guide to Testing Statistical Significance in NLP
- Confusion matrices: *J&M3 p484*