Evaluation in Natural Language Processing

Lecture at ESSLLI 2022

DOI: 10.5281/zenodo.6667766

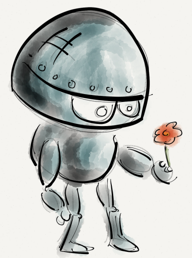
Outline

Section 1

Introduction

Introduction

- Once a model is obtained, it is crucial to study its performance and impact
- How do we find a correlation between quality and evaluation score?
- What are the common techniques in Natural Language Processing (NLP)?
- We need reproducibility, scalability, and proper benchmarking (Dacrema et al., 2019)



Source: bamenny (2016)

Core Idea: Measure Twice and Cut Once

You can invent a method every day. How do you know if it is actually good?

How to Evaluate?

Online Evaluation

Pros:

+ Objective
+ Interpretable

Cons:

- Can hurt users
- Irreproducible
- Poor scalability

Offline Evaluation

Pros:

+ Scalable
+ Reproducible
+ Safe

Cons:

- Can be unobjective

Today we will focus on **offline evaluation**,
refer to Kohavi et al. (2020) on online evaluation.

Ground Truth

Offline evaluation requires **ground truth** to be available; typical sources are:

- Expert Assessment
- Gold and Silver Standards
- Crowdsourcing

Source: Finnsson (2017)

Ground Truth: Expert Assessment I

In **Expert Assessment**, the output of the system is manually evaluated by a group of expert assessors who ultimately decide whether it works well or not.

Examples:

- Search engines
- Sensitive domains (Medicine, Security, etc.)

Ground Truth: Expert Assessment II

Pros:

- Very high quality and accuracy
- Evaluation can be very complex

Cons:

- Does not scale
- Have to trust the experts
- Only one data point per expert

Ground Truth: Gold and Silver Standards I

Gold Standards are well-known, expert-annotated, and trustworthy datasets dedicated to a particular problem. **Silver Standards** are the gold ones matched with unverified data.

Examples:

- **Gold:** Penn Treebank (Marcus et al., 1993), WordNet (Fellbaum, 1998), FrameNet (Baker et al., 1998)
- **Silver:** BabelNet (Navigli et al., 2012)

Ground Truth: Gold and Silver Standards II

Pros:

- + Very high quality and accuracy
- + Trusted by the community

Cons:

- − Could be missing for your task or be smaller than needed
- − Requires expert annotation or matching

Ground Truth: Crowdsourcing I

Crowdsourcing is a type of participative *online activity* in which *the requester* proposes to *a group of individuals* ... the voluntary undertaking of *a task* (Estellés-Arolas et al., 2012).

Examples:

- **Data Acquisition:** Wikipedia, Wiktionary, ESP Game (von Ahn et al., 2004), Common Voice (Ardila et al., 2020)
- **System Evaluation:** Search Relevance (Alonso et al., 2008), Machine Translation (Callison-Burch, 2009), Intruders (Chang et al., 2009)

Ground Truth: Crowdsourcing II

Pros:

+ Scalability
+ Flexibility

Cons:

- Need for task design
- Need for quality control

Decision Support Systems

Suppose that you have a *decision support system* (DSS).

- The system's response can be positive or negative; both can be true or false:
  Type I error *aka* false positive (FP)
  Type II error *aka* false negative (FN)
- A **confusion matrix** $C \in {\mathbb{Z}^{0+}}^{k \times k}$ shows how well a *decision support system* works for $k > 1$ classes
- ! It would be more convenient to have a single number indicating the system's performance

		Actual $+$	Actual $-$
Predicted	$+$	TP	FP
Predicted	$-$	FN	TN

Note that in some sources this matrix is transposed!

Information Retrieval Evaluation

Two ways for evaluating Information Retrieval (IR) systems: **unranked** and **ranked**, see van Rijsbergen (1979, Chapter 7) and Manning et al. (2008, Chapter 8).

In *unranked evaluation*, a set of all the obtained results is assessed.

- Accuracy, Precision, Recall, and F-score, Fowlkes–Mallows Index, ROC-AUC, …

In *ranked evaluation*, an ordered list of results is assessed.

- Precision@K, Mean Average Precision, NDCG, pFound and ERR, …

Section 2

Classification Evaluation

Accuracy

Accuracy (Ac) is the fraction of correct responses provided by the system.

$$\mathrm{Ac} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

- Interpretable
- Easy to compare against a random baseline of $\mathrm{Ac} = \frac{1}{k}$
- Biased when the class distribution is skewed (Powers, 2008)

Precision and Recall

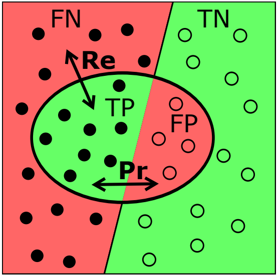
Kent et al. (1955) designed precision and recall for IR systems.

Precision (Pr) is the fraction of retrieved documents that are *relevant*:

$$\mathrm{Pr} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Recall (Re) is the fraction of relevant documents that are *retrieved*:

$$\mathrm{Re} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$



Source: Nichtich (2008)

- Not very useful without each other
- Biased when the class distribution is skewed (Powers, 2008)
- How to get a single-figure measure?

F-score (*aka* F-measure or Dice coefficient)

F-score (F_β) is the weighted harmonic mean of precision and recall (van Rijsbergen, 1979), also known as the Dice coefficient:

$$\mathrm{F}_\beta = (1 + \beta^2) \cdot \frac{\mathrm{Pr} \cdot \mathrm{Re}}{\beta^2 \cdot \mathrm{Pr} + \mathrm{Re}} \qquad \mathrm{F}_1 = 2 \cdot \frac{\mathrm{Pr} \cdot \mathrm{Re}}{\mathrm{Pr} + \mathrm{Re}}$$

Fowlkes–Mallows Index (FM) is the geometric mean of precision and recall (Fowlkes et al., 1983):

$$\mathrm{FM} = \sqrt{\mathrm{Pr} \cdot \mathrm{Re}}$$

So far we considered only the binary classification case.

Multiple Classes

What if we have more than two classes, i.e., $k > 2$?

Micro-Average. Compute scores for each class together:

$$\mathrm{Pr}_{\text{micro}} = \frac{\sum_{i=1}^{k} \mathrm{TP}_i}{\sum_{i=1}^{k}(\mathrm{TP}_i + \mathrm{FP}_i)}, \qquad \mathrm{Re}_{\text{micro}} = \frac{\sum_{i=1}^{k} \mathrm{TP}_i}{\sum_{i=1}^{k}(\mathrm{TP}_i + \mathrm{FN}_i)}$$

Macro-Average. Compute Pr_i and Re_i for each $1 \leq i \leq k$:

$$\mathrm{Pr}_{\text{macro}} = \frac{1}{k} \sum_{i=1}^{k} \mathrm{Pr}_i\,, \qquad \mathrm{Re}_{\text{macro}} = \frac{1}{k} \sum_{i=1}^{k} \mathrm{Re}_i$$

Weighted. For each $1 \leq i \leq k$ use the number of gold instances $\#(i)$:

$$\mathrm{Pr}_{\text{weighted}} = \frac{\sum_{i=1}^{k}(\#(i) \cdot \mathrm{Pr}_i)}{\sum_{i=1}^{k} \#(i)}, \qquad \mathrm{Re}_{\text{weighted}} = \frac{\sum_{i=1}^{k}(\#(i) \cdot \mathrm{Re}_i)}{\sum_{i=1}^{k} \#(i)}$$

Try not to use averaging, but if necessary,
use macro-average (Gösgens et al., 2021b).

Issues with Traditional IR Scores

Despite the huge popularity of $\mathrm{Ac}, \mathrm{Pr}, \mathrm{Re}$, etc., these scores have major issues (Powers, 2008; Chicco et al., 2020; Gösgens et al., 2021b):

- they are biased toward dominant classes
- they can easily be manipulated
- they are not *metrics*

Source: Rahman Rony (2016)

Bias

Consider a part-of-speech tagger that classifies everything as NN and our evaluation dataset is imbalanced.

$$\mathrm{Ac} = \frac{90}{90 + 10 + 0 + 0} = 90\%$$

$$\mathrm{Pr} = \frac{90}{90 + 10} = 90\%$$

$$\mathrm{Re} = \frac{90}{90 + 0} = 100\%$$

$$\mathrm{F}_1 = 2 \cdot \frac{0.9 \cdot 1}{0.9 + 1} \approx 95\%$$

$$\mathrm{FM} = \sqrt{0.9 \cdot 1} \approx 95\%$$

P\E	NN	VBP
NN	90	10
VBP	0	0

Not a very good evaluation of such a trivial classifier.

Labels are part-of-speech (PoS) tags from the Penn Treebank (Marcus et al., 1993), e.g., `influence/NN` is a singular or mass *noun*, `influence/VBP` is a non-third person singular present *verb*.

Mathews Correlation Coefficient

Matthews (1975) proposed the correlation coefficient $\mathrm{MCC} \in [-1, 1]$ that balances classes of different sizes:

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}$$

In the previous example, $\mathrm{MCC} = \frac{90 \times 0 - 10 \times 0}{\sqrt{(90+10)(90+0)(0+10)(0+0)}} = 0$.

Mathews Correlation Coefficient: Multiclass

Gorodkin (2004) generalized MCC to multiple classes as the R_K coefficient of the confusion matrix C:

$$R_K = \frac{\sum_{k,l,m} C_{kk}C_{lm} - C_{kl}C_{mk}}{\sqrt{\sum_k \left(\sum_l C_{kl}\right)\left(\sum_{\substack{l' \\ k' \neq k}} C_{k'l'}\right)}\sqrt{\sum_k \left(\sum_l C_{lk}\right)\left(\sum_{\substack{l' \\ k' \neq k}} C_{l'k'}\right)}}$$

MCC and R_K are stable except in very extreme cases,
see Chicco et al. (2020) and Gösgens et al. (2021b) for a detailed discussion.

Symmetric Balanced Accuracy

Gösgens et al. (2021b) proposed **Symmetric Balanced Accuracy** (SBA), addressing many drawbacks of the previous criteria. Given the confusion matrix C and the number of classes $k \geq 2$, we define it as

$$\mathrm{SBA} = \frac{1}{2k} \sum_{i=1}^{k} \left(\frac{C_{ii}}{a_i} + \frac{C_{ii}}{b_i} \right),$$

where a_i is the number of actual instances and b_i is the number of predicted instances for i-th class; the total number of instances is $n = \sum_{i=1}^{k} \sum_{j=1}^{k} C_{ij}$.

Instances for some classes might be missing,
so if $a_i = 0$, $\frac{C_{ii}}{a_i}$ is replaced with $\frac{b_i}{n}$, and if $b_i = 0$, $\frac{C_{ii}}{b_i}$ is replaced with $\frac{a_i}{n}$.

In the last example,
$\mathrm{SBA} = \frac{1}{2 \times 2} \left(\boxed{\frac{90}{90} + \frac{90}{100}} + \boxed{\frac{0}{10} + \frac{10}{100}} \right) = 0.5.$

Classification Curves

- A single number is not enough: it is important to study the algorithm's sensitivity and specificity
- Receiver Operating Characteristics (ROC) and Precision-Recall (PR) curves allow examining these properties
- ! They can be applied as soon as the method returns the probability, confidence, or decision value, etc.

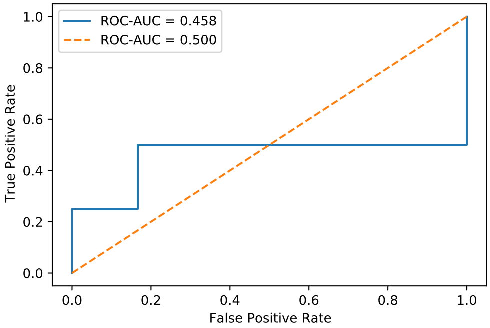
Source: rawpixel (2017)

Receiver Operating Characteristics

Receiver Operating Characteristics (ROC) curve shows a trade-off between true positive rate (recall) and false positive rate, $\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}$.

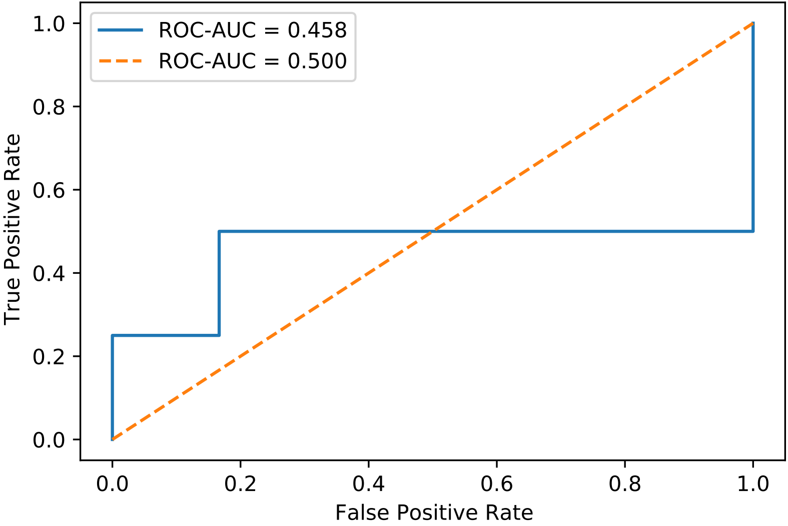
1. Perform the classification and obtain a score for each response
2. Iterate over the scores and plot FPR and TPR points
3. Compute the area under curve ($\mathrm{ROC\text{-}AUC}$) using the trapezoidal rule

! $\mathrm{ROC\text{-}AUC} = 0.5$ is a random classifier baseline



Consider using the more informative precision-recall (PR) curve (Saito et al., 2015).

Receiver Operating Characteristics: Example



This is an example following Manning et al. (2008, Section 8.4)

Classification Evaluation: Wrap-Up

- Always check for class imbalance
- Use the MCC, SBA, and $\mathrm{ROC\text{-}AUC}$ measures to report quality
- Report a PR curve to evaluate the precision and recall dynamics (we will discuss it today later)
- **Implementations:** R, scikit-learn (Pedregosa et al., 2011) for Python, etc.

Source: Free-Photos (2016)

Section 3

Clustering Evaluation

Clustering Evaluation

- Two classes of clustering evaluation criteria: internal and external
- **Internal criteria** measure intra-cluster similarity and inter-cluster similarity, which do not necessarily correspond to your task (Manning et al., 2008, Chapter 16)
- **External criteria** compare the obtained clustering with ground truth; see discussion on measures in Yang et al. (2013, Section 6.2) and Gösgens et al. (2021a)

Source: Buissinne (2016)

Pairwise Evaluation

- A set of objects V can be transformed into a complete graph (V, E) with $|E| = \binom{|V|}{2}$ undirected edges, and we can perform the same operation for every cluster of V
- Union of cluster element pairs $P \subseteq V^2$ can be compared to the union of gold cluster element pairs $P_G \subseteq V^2$ using *paired F-score* (Manandhar et al., 2010):

$$\mathrm{TP} = |P \cup P_G|, \quad \mathrm{FP} = |P \setminus P_G|, \quad \mathrm{FN} = |P_G \setminus P|$$
$$\mathrm{Pr} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \quad \mathrm{Re} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad \mathrm{F}_1 = 2\frac{\mathrm{Pr} \cdot \mathrm{Re}}{\mathrm{Pr} + \mathrm{Re}}$$

- This approach is interpretable and allows applying the classification evaluation techniques
- It does not explicitly assess the quality of overlapping clusters (larger are preferred)

Adjusted Rand Index

- Rand (1971) proposed an index for clustering evaluation that is the same as the accuracy measure Ac from the classification evaluation:

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

- Hubert et al. (1985) proposed a chance-corrected version, **Adjusted Rand Index**:

$$\mathrm{ARI} = \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i\cdot}}{2} + \sum_j \binom{n_{\cdot j}}{2}\right] - \left[\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}\right] / \binom{n}{2}},$$

where n_{ij} is a contingency table

Purity

Purity is a measure of the extent to which clusters contain a single class, which is useful for evaluating a *partitioning* C against the gold partitioning C_G (Manning et al., 2008):

$$\mathrm{PU} = \frac{1}{|C|} \sum_{i}^{|C|} \max_{j} |C^i \cap C_G^j|$$

$$\mathrm{iPU} = \frac{1}{|C_G|} \sum_{j}^{|C_G|} \max_{i} |C^i \cap C_G^j|$$

$$\mathrm{F}_1 = 2\frac{\mathrm{PU} \cdot \mathrm{iPU}}{\mathrm{PU} + \mathrm{iPU}}$$

Normalized Modified Purity

Kawahara et al. (2014) proposed *normalized modified purity* for *soft* clustering that considers weighted overlaps $\delta_{C^i}(C^i \cap C_G^j)$:

$$\mathrm{nmPU} = \frac{1}{|C|} \sum_{i \text{ s.t. } |C^i|>1}^{|C|} \max_{1 \leq j \leq |C_G|} \delta_{C^i}(C^i \cap C_G^j)$$

$$\mathrm{niPU} = \frac{1}{|C_G|} \sum_{j=1}^{|G|} \max_{1 \leq i \leq |C|} \delta_{C_G^j}(C^i \cap C_G^j)$$

$$\mathrm{F}_1 = 2\frac{\mathrm{nmPU} \cdot \mathrm{niPU}}{\mathrm{nmPU} + \mathrm{niPU}}$$

Normalized Modified Purity: Example

Actual	$\{\text{bank} : 1\}$,
	$\{\text{riverbank} : 1, \text{streambank} : 1, \text{streamside} : 1\}$,
	$\{\text{building} : 1, \text{bank building} : 1\}$
Predicted	$\{\text{bank} : 0.5, \text{riverbank} : 1, \text{streambank} : 1, \text{streamside} : 1\}$,
	$\{\text{bank} : 0.5, \text{building} : 1, \text{bank building} : 1\}$

$$\begin{aligned}\mathrm{nmPU} &= 0.833\\ \mathrm{niPU} &= 1.000\\ \mathrm{F}_1 &= 0.909\end{aligned}$$

This is an example from Ustalov et al. (2019)

Clustering Evaluation: Wrap-Up

- Evaluate hard clustering with ARI and soft clustering with $\mathrm{nmPU/niPU}$
- More difficult tasks, such as taxonomy evaluation, can be reduced to clustering evaluation (Velardi et al., 2013)
- **Implementations:** scikit-learn (Pedregosa et al., 2011), xmeasures (Lutov et al., 2019), watset-java (Ustalov et al., 2019), etc.

Source: Pexels (2016)

Section 4

Ranked Evaluation

- Assume we have retrieved top $k \in \mathbb{N}$ results
- We want the most relevant items to be on the top of this list
- Measures include binary ($\Pr@k$, MAP, MRR) and graded (NDCG, $\mathrm{pFound/ERR}$), etc.

Source: Amos (2011)

Average Precision

Precision@k ($\Pr@k$) is the fraction of relevant items in the k top retrieved items for the given query:

$$\Pr@k = \sum_{i=1}^{k} \mathbf{1}_{i\text{-th item is relevant}}$$

Average Precision (AP) is the non-interpolated area under the PR curve (Buckley et al., 2000):

$$\mathrm{AP} = \frac{1}{\text{\# of relevant items}} \sum_{i=1}^{k} \Pr@i \cdot \mathbf{1}_{i\text{-th item is relevant}}$$

Mean Average Precision (MAP) is the average AP of all the queries Q:

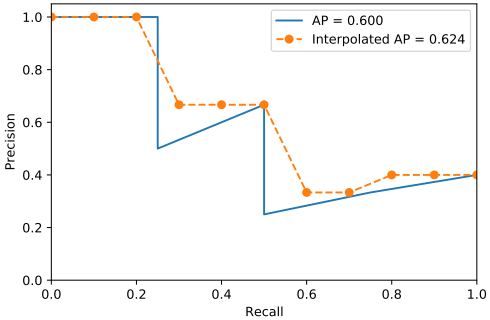
$$\mathrm{MAP} = \frac{1}{|Q|} \sum_{q \in Q} \mathrm{AP}(q)$$

Precision-Recall Curve

Precision-Recall (PR) curve shows a trade-off between precision and recall.

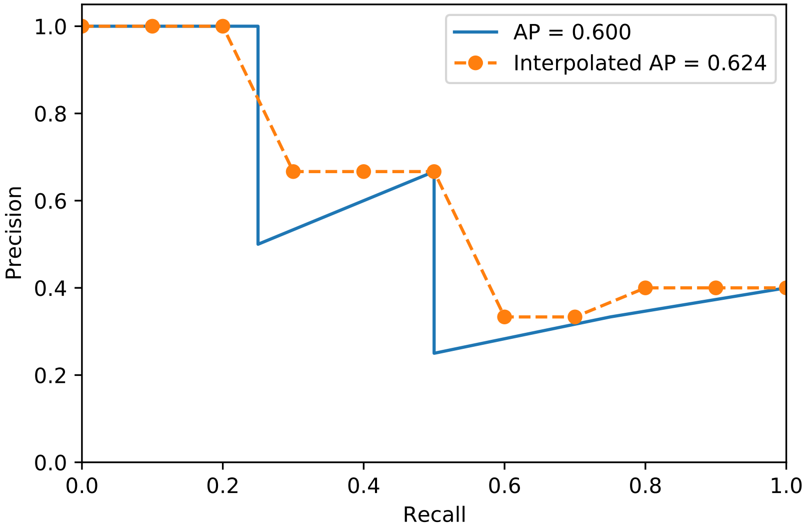
1. Perform the classification and obtain a score for each response
2. Compute precision and recall at each level k as well as average precision
3. Compare systems using an *11-point interpolated PR curve*

! Due to the interpolation, $\mathrm{PR\text{-}AUC}$ might be too optimistic; compute the average precision (AP)



If one method dominates another on ROC, it will dominate on PR, too (Davis et al., 2006).

Precision-Recall Curve: Example



This is an example following Manning et al. (2008, Section 8.4)

Normalized Discounted Cumulative Gain

Cumulative Gain (CG) in top k items is a sum of the relevance grades $\mathrm{rel}_i \in \mathbb{N}$ corresponding to every i-th retrieved item (Järvelin et al., 2002; Wang et al., 2013):

$$\mathrm{CG} = \sum_{i=1}^{k} \mathrm{rel}_i$$

Discounted Cumulative Gain (DCG) is a CG divided by the logarithm of each item's position:

$$\mathrm{DCG} = \mathrm{rel}_1 + \sum_{i=2}^{k} \frac{\mathrm{rel}_i}{\log_2 i}$$

Normalized Discounted Cumulative Gain (NDCG) is the fraction of the obtained DCG in the "perfect" DCG:

$$\mathrm{NDCG} = \frac{\mathrm{DCG}}{\text{ideal}\ \mathrm{DCG}}$$

Yandex' pFound

pFound is a cascade probabilistic ranked evaluation measure that simulates how a user looks at the search results.

The user looks at items sequentially in top-down order and stops if either the relevant item is found or they gave up with probability pBreak.

$$\mathrm{pFound} = \sum_{i=1}^{n} \overbrace{\mathrm{pLook}[i]}^{\substack{\text{user looks}\\ \text{at } i\text{-th item}}} \cdot \overbrace{\mathrm{pRel}[i]}^{\substack{i\text{-th item}\\ \text{is relevant}}}$$

$$\mathrm{pLook}[i] = \begin{cases} 1, & i = 1 \\ \mathrm{pLook}[i-1] \cdot (1 - \mathrm{pRel}[i-1]) \cdot (1 - \mathrm{pBreak}), & i \neq 1 \end{cases}$$

$$\mathrm{pBreak} = 0.15$$

Invented at Yandex and was the optimization goal back in 2007 (Segalovich, 2010); similar to the Expected Reciprocal Rank (Chapelle et al., 2009, Section 7.2).

Expected Reciprocal Rank

Mean Reciprocal Rank (MRR) is the mean rank position of the first relevant item (rank) in all the queries Q (Voorhees, 1999):

$$\mathrm{MRR} = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{\mathrm{rank}_q}$$

Expected Reciprocal Rank (ERR) is the expected reciprocal length of time that the user will take to find a relevant document (Chapelle et al., 2009)

$$\mathrm{ERR} = \sum_{r=1}^{n} \frac{1}{r} \left(\prod_{i=1}^{r-1} (1 - R_i) \cdot R_r \right)$$

To translate relevance grades to the probability of relevance, we define $\mathcal{R}_g : g \to [0, 1], \forall g \in \{0, \ldots, g_{\max}\}$ and then compute the score:

$$\mathcal{R}_g = \frac{2^g - 1}{2^{g_{\max}}}$$

Expected Reciprocal Rank: Algorithm

Input: relevance grades $g_r, 1 \leq r \leq n$, mapping $R : g_r \to [0, 1]$
Output: expected reciprocal rank ERR

$p \leftarrow 1$
$\mathrm{ERR} \leftarrow 0$
for $r \leftarrow 1 \ldots n$ **do**
    $R \leftarrow \mathcal{R}(g_r)$
    $\mathrm{ERR} \leftarrow \mathrm{ERR} + p \cdot \dfrac{R}{r}$
    $p \leftarrow p \cdot (1 - R)$
return ERR

Explicit Reciprocal Rank: Example

d	g	R
4	0	0
5	0	0
9	2	0.375
2	0	0
1	3	0.875
8	0	0
10	1	0.125
6	0	0
7	0	0
3	1	0.125

$$g_{\max} = 3$$

$$\mathrm{ERR} = 0.237$$

This is an example following Manning et al. (2008, Section 8.4)

Expected Reciprocal Rank: Discussion

Pros:

- \+ Sound method that takes into account user behaviour
- \+ Fast; running time is $O(n)$

Cons:

- − Model assumptions need to be met
- − Low discriminative power (Sakai, 2006)

Ranked Evaluation: Wrap-Up

- Use MAP for binary relevance, NDCG for graded relevance, and pFound/ERR for graded relevance with the user's behaviour
- Usually, one has to limit the number of top-k documents, see discussion in Wang et al. (2013)
- **Implementations:** scikit-learn (Pedregosa et al., 2011), RankEval (Lucchese et al., 2017)

Source: Dumlao (2017)

Section 5

Significance and Reliability

Significance and Reliability

- How to determine if the method is not just good, but outperforms other approaches?
- How to ensure the reliability of expert or crowd responses?
- In this section we will discuss computational techniques for *statistical significance testing* and *inter-rater reliability analysis*

Source: Merrill (2014)

Statistical Testing

We state two hypotheses, *null* and *alternative*, and use a statistical test to determine whether to reject the null hypothesis or not.

There has been an active discussion on the choice of statistical tests in IR and NLP (Smucker et al., 2007; Dror et al., 2018):

- some tests assume normally-distributed data: Z-test, *t*-test
- some do not have enough statistical power: Wilcoxon signed-rank test, sign test, etc.
- some were not feasible in the past: randomization test and bootstrap

Following the recommendations in Yeh (2000) and Smucker et al. (2007), we will apply the **randomization test** (*aka* permutation test): "no difference after *shuffling*".

Randomization Test: Algorithm

Input: vectors $\vec{A}$ and $\vec{B}$ such that $|\vec{A}| = |\vec{B}|$,
number of trials $N \in \mathbb{N}$, quality criterion $f : \mathbb{R}^{|\vec{A}|} \to \mathbb{R}$
Output: two-tailed p-value

1: $\mathrm{uncommon} \leftarrow \{1 \leq i \leq |\vec{A}| : A_i \neq B_i\}$
2: $s \leftarrow 0$
3: **for** $n \leftarrow 1 \ldots N$ **do**
4: $\quad \vec{A}', \vec{B}' \leftarrow \vec{A}, \vec{B}$ ▷ Copy $\vec{A}$ and $\vec{B}$
5: $\quad$ **for all** $i \in \mathrm{uncommon}$ **do**
6: $\quad\quad$ **if** $\mathtt{random}(\{0, 1\}) = 0$ **then** ▷ Flip a coin
7: $\quad\quad\quad A'_i, B'_i \leftarrow B_i, A_i$ ▷ Shuffle by swapping the values if tails
8: $\quad$ **if** $|f(\vec{A}') - f(\vec{B}')| \geq |f(\vec{A}) - f(\vec{B})|$ **then**
9: $\quad\quad \mathrm{s} \leftarrow \mathrm{s} + 1$ ▷ Note that we evaluate the absolute difference
10: **return** $\dfrac{s}{N}$ ▷ This value can be compared to a significance level

This technique can be used with mean, F-score, and other quality criteria (Yeh, 2000).

Randomization Test: Example

Example from Padó (2006) with $f = \mathrm{mean}$

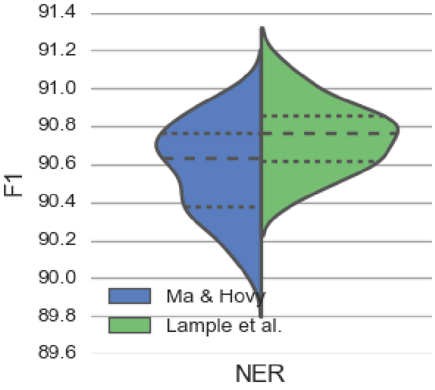
$\vec{A} = (1, 2, 1, 2, 2, \mathbf{2}, 0)$ $\qquad$ $\mathrm{mean}(\vec{A}) \approx 1.4286$
$\vec{B} = (4, 5, 5, 4, 3, \mathbf{2}, 1)$ $\qquad$ $\mathrm{mean}(\vec{B}) \approx 3.4286$

The uncommon elements are $\{1, 2, 3, 4, 5, 7\}$ and the difference in means is $|\mathrm{mean}(\vec{A}) - \mathrm{mean}(\vec{B})| = 2$.

Having performed $N = 10^6$ iterations, we obtain $p \approx 0.0313$, which is, given the significance level of 0.05, suggesting a statistically significant difference.

Statistical Testing: Discussion

- Always perform statistical testing and report not only statistical significance but also the score distributions (Reimers et al., 2017)
- The topic is huge and deserves a dedicated course; see more in the context of NLP in Dror et al. (2018)



Source: Reimers et al. (2017)

Inter-Rater Agreement

- How *reliable* is the annotation?
- In the example in 51.1% of cases the raters agree with each other, is it a good thing?
- A low value indicates issues with task design and difficulty: *the answers might make no sense*

	w_1	w_2	w_3	w_4
t_1	NN		NN	NN
t_2	NN	VBP	VBP	NN
t_3	VBP	VBP	VBP	NN
t_4	VBP	NN	NN	VBP

Krippendorff's α (2018) is a versatile inter-rater agreement measure that takes into account the *observed* disagreement D_o and the *expected* disagreement D_e:

$$\alpha = 1 - \frac{D_o}{D_e}$$

α is chance-corrected, handles missing values, and allows for arbitrary distance functions (binary, nominal, interval, etc.)

In the *nominal* case of C classes, α is computed using a coincidence matrix $O \in \mathbb{R}^{|C|\times|C|}$:

$$_{\text{nominal}}\alpha = 1 - (n-1)\frac{n - \sum_{c \in C} O_{cc}}{n^2 - \sum_{c \in C} n_c^2},$$

where $n_c = \sum_{k \in C} O_{ck}$ and $n = \sum_{c \in C} n_c$.

Krippendorff's α: Algorithm

Input: m raters, N tasks, set of classes C, data matrix $U \in (\{-\} \cup C)^{m \times N}$ ▷ Missing values are $(-)$

Output: $0 \leq {}_{\text{nominal}}\alpha \leq 1$

1: $O_{ck} \leftarrow 0$ **for all** $c \in C, k \in C$
2: **for** $u \in 1 \ldots N$ **do** ▷ Each task
3: **for all** $c, k \in P(U_u^\top, 2)$ **do** ▷ Each possible non-missing (c, k) pair
4: $O_{ck} \leftarrow O_{ck} + \frac{1}{m_u - 1}$ ▷ m_u is the number of raters in task u
5: $n_c \leftarrow \sum_{k \in C} O_{ck}$ **for all** $c \in C$
6: $n \leftarrow \sum_{c \in C} n_c$
7: **return** $1 - (n - 1)\dfrac{n - \sum_{c \in C} O_{cc}}{n^2 - \sum_{c \in C} n_c^2}$

Krippendorff's α: Example

$$O = \begin{pmatrix} 4.33 & 3.67 \\ 3.67 & 3.33 \end{pmatrix}$$

$$n_c = \begin{pmatrix} 8 & 7 \end{pmatrix}$$

$$n = 15$$

$\mathsf{U}^\top$

	w_1	w_2	w_3	w_4
t_1	NN		NN	NN
t_2	NN	VBP	VBP	NN
t_3	VBP	VBP	VBP	NN
t_4	VBP	NN	NN	VBP

$$_{\text{nominal}}\alpha = 1 - (n-1)\frac{n - \sum_{c \in C} O_{cc}}{n^2 - \sum_{c \in C} n_c^2} = 1 - 14\frac{15 - (4.33 + 3.33)}{15^2 - (8^2 + 7^2)}$$

$$= 1 - \frac{102.76}{112} \approx 0.083$$

Krippendorff's α: Discussion

- **Interpretation** by Krippendorff (2018):

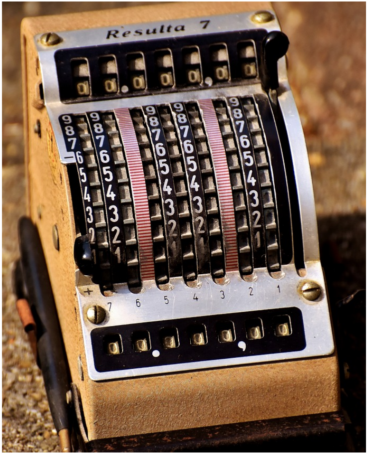
  $\alpha \geq 0.800$: reliable annotation (reliability $\not\Rightarrow$ correctness!)

  $0.667 \leq \alpha < 0.800$: tentative conclusions only
- **Implementations:** DKPro for Java (Meyer et al., 2014), NLTK for Python (Bird et al., 2017), irr for R, etc.
- Computing α is complex and slow; resampling and bootstrap might be useful on large datasets

Source: rawpixel (2018)

Significance and Reliability: Wrap-Up

- Trust, but verify: always evaluate and report whether your results are significant and your labels are reliable
- Significance can be evaluated using a permutation test; see more in Smucker et al. (2007) and Dror et al. (2018)
- Reliability can be evaluated using a convenient single number, Krippendorff's α; see a good overview in Artstein et al. (2008)



Source: Alexas_Fotos (2017)

Section 6

Conclusion

Conclusion

- Machine Learning incurs massive maintenance costs (Sculley et al., 2015), so the effect should be carefully analyzed and evaluated
- Choose quality criteria wisely, compare the results against those of others, and perform statistical testing
- Sometimes the dataset is very large, so recall can be only *estimated* on a smaller sample
- Not covered here: behavioural testing (Ribeiro et al., 2020), taxonomy evaluation (Bordea et al., 2016) and other evaluation tasks, and much more

Source: shbs (2017)

Questions?

Contacts

Dr. **Dmitry Ustalov**

https://github.com/dustalov

mailto:dmitry.ustalov@gmail.com

0000-0002-9979-2188

Revision: 5d35748

References I

von Ahn L. and Dabbish L. (2004). Labeling Images with a Computer Game. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '04. Vienna, Austria: ACM, pp. 319–326. DOI: 10.1145/985692.985733.

Alonso O., Rose D. E., and Stewart B. (2008). Crowdsourcing for Relevance Evaluation. *SIGIR Forum*, vol. 42, no. 2, pp. 9–15. DOI: 10.1145/1480506.1480508.

Ardila R. et al. (2020). Common Voice: A Massively-Multilingual Speech Corpus. *Proceedings of The 12th Language Resources and Evaluation Conference*. LREC 2020. Marseille, France: European Language Resources Association (ELRA), pp. 4218–4222. URL: https://aclanthology.org/2020.lrec-1.520.

Artstein R. and Poesio M. (2008). Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, vol. 34, no. 4, pp. 555–596. DOI: 10.1162/coli.07-034-R2.

Baker C. F., Fillmore C. J., and Lowe J. B. (1998). The Berkeley FrameNet Project. *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*. ACL '98/COLING '98. Montréal, QC, Canada: Association for Computational Linguistics, pp. 86–90. DOI: 10.3115/980845.980860.

Bird S., Klein E., and Loper E. (2017). Natural Language Processing with Python. 2nd Edition. O'Reilly Media. ISBN: 978-1-4919-1342-0. URL: https://www.nltk.org/book/.

Bordea G., Lefever E., and Buitelaar P. (2016). SemEval-2016 Task 13: Taxonomy Extraction Evaluation (TExEval-2). *Proceedings of the 10th International Workshop on Semantic Evaluation*. SemEval-2016. San Diego, CA, USA: Association for Computational Linguistics, pp. 1081–1091. DOI: 10.18653/v1/S16-1168.

Buckley C. and Voorhees E. M. (2000). Evaluating Evaluation Measure Stability. *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '00. Athens, Greece: Association for Computing Machinery, pp. 33–40. DOI: 10.1145/345508.345543.

Callison-Burch C. (2009). Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. EMNLP 2009. Singapore: Association for Computational Linguistics and Asian Federation of Natural Language Processing, pp. 286–295. DOI: 10.3115/1699510.1699548.

Chang J. et al. (2009). Reading Tea Leaves: How Humans Interpret Topic Models. *Advances in Neural Information Processing Systems 22*. NIPS 2009. Vancouver, BC, Canada: Curran Associates, Inc., pp. 288–296. URL: https://papers.nips.cc/paper/3700-reading-tea-leaves-how-humans-interpret-topic-models.pdf.

Chapelle O. et al. (2009). Expected Reciprocal Rank for Graded Relevance. *Proceedings of the 18th ACM Conference on Information and Knowledge Management*. CIKM '09. Hong Kong, China: Association for Computing Machinery, pp. 621–630. DOI: 10.1145/1645953.1646033.

Chicco D. and Jurman G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, vol. 21, no. 1, p. 6. DOI: 10.1186/s12864-019-6413-7.

References II

Dacrema M. F., Cremonesi P., and Jannach D. (2019). Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches. *Proceedings of the 13th ACM Conference on Recommender Systems*. RecSys '19. Copenhagen, Denmark: Association for Computing Machinery, pp. 101–109. DOI: 10.1145/3298689.3347058.

Davis J. and Goadrich M. (2006). The Relationship between Precision-Recall and ROC Curves. *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. Pittsburgh, PA, USA: Association for Computing Machinery, pp. 233–240. DOI: 10.1145/1143844.1143874.

Dror R. et al. (2018). The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. ACL 2018. Melbourne, VIC, Australia: Association for Computational Linguistics, pp. 1383–1392. DOI: 10.18653/v1/P18-1128.

Estellés-Arolas E. and González-Ladrón-de-Guevara F. (2012). Towards an integrated crowdsourcing definition. *Journal of Information Science*, vol. 38, no. 2, pp. 189–200. DOI: 10.1177/0165551512437638.

Fellbaum C. (1998). WordNet: An Electronic Database. Massachusetts, MA, USA: MIT Press. ISBN: 978-0-262-06197-1. DOI: 10.7551/mitpress/7287.001.0001.

Fowlkes E. B. and Mallows C. L. (1983). A Method for Comparing Two Hierarchical Clusterings. *Journal of the American Statistical Association*, vol. 78, no. 383, pp. 553–569. DOI: 10.1080/01621459.1983.10478008.

Gorodkin J. (2004). Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry*, vol. 28, no. 5, pp. 367–374. DOI: 10.1016/j.compbiolchem.2004.09.006.

Gösgens M., Tikhonov A., and Prokhorenkova L. (2021a). Systematic Analysis of Cluster Similarity Indices: How to Validate Validation Measures. *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. Proceedings of Machine Learning Research. Online: PMLR, pp. 3799–3808. URL: https://proceedings.mlr.press/v139/gosgens21a/gosgens21a.pdf.

Gösgens M. et al. (2021b). Good Classification Measures and How to Find Them. *Advances in Neural Information Processing Systems 34*. NeurIPS 2021. Online: Curran Associates, Inc., pp. 17136–17147. URL: https://proceedings.neurips.cc/paper/2021/file/8e489b4966fe8f703b5be647f1cbae63-Paper.pdf.

Hubert L. and Arabie P. (1985). Comparing partitions. *Journal of Classification*, vol. 2, no. 1, pp. 193–218. DOI: 10.1007/BF01908075.

Järvelin K. and Kekäläinen J. (2002). Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems*, vol. 20, no. 4, pp. 422–446. DOI: 10.1145/582415.582418.

Kawahara D., Peterson D. W., and Palmer M. (2014). A Step-wise Usage-based Method for Inducing Polysemy-aware Verb Classes. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics Volume 1: Long Papers*. ACL 2014. Baltimore, MD, USA: Association for Computational Linguistics, pp. 1030–1040. DOI: 10.3115/v1/P14-1097.

References III

Kent A. et al. (1955). Machine literature searching VIII. Operational criteria for designing information retrieval systems. *American Documentation*, vol. 6, no. 2, pp. 93–101. DOI: 10.1002/asi.5090060209.

Kohavi R., Tang D., and Xu Y. (2020). Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing. 1st edition. Cambridge University Press. ISBN: 978-1-108-72426-5. DOI: 10.1017/9781108653985. URL: https://experimentguide.com/.

Krippendorff K. (2018). Content Analysis: An Introduction to Its Methodology. Fourth Edition. Thousand Oaks, CA, USA: SAGE Publications, Inc. ISBN: 978-1-5063-9566-1.

Lucchese C. et al. (2017). RankEval: An Evaluation and Analysis Framework for Learning-to-Rank Solutions. *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '17. Shinjuku, Tokyo, Japan: Association for Computing Machinery, pp. 1281–1284. DOI: 10.1145/3077136.3084140.

Lutov A., Khayati M., and Cudré-Mauroux P. (2019). Accuracy Evaluation of Overlapping and Multi-Resolution Clustering Algorithms on Large Datasets. *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*. Kyoto, Japan: IEEE, pp. 1–8. DOI: 10.1109/BIGCOMP.2019.8679398.

Manandhar S. et al. (2010). SemEval-2010 Task 14: Word Sense Induction & Disambiguation. *Proceedings of the 5th International Workshop on Semantic Evaluation*. SemEval 2010. Uppsala, Sweden: Association for Computational Linguistics, pp. 63–68. URL: https://aclanthology.org/S10-1011.

Manning C. D., Raghavan P., and Schütze H. (2008). Introduction to Information Retrieval. Cambridge University Press. ISBN: 978-0-521-86571-5. URL: https://nlp.stanford.edu/IR-book/.

Marcus M. P., Santorini B., and Marcinkiewicz M. A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, vol. 19, no. 2, pp. 313–330. URL: https://aclanthology.org/J93-2004.

Matthews B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, vol. 405, no. 2, pp. 442–451. DOI: 10.1016/0005-2795(75)90109-9.

Meyer C. M. et al. (2014). DKPro Agreement: An Open-Source Java Library for Measuring Inter-Rater Agreement. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*. COLING 2014. Dublin, Ireland: Dublin City University and Association for Computational Linguistics, pp. 105–109. URL: https://aclanthology.org/C14-2023.

Navigli R. and Ponzetto S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, vol. 193, pp. 217–250. DOI: 10.1016/j.artint.2012.07.001.

Padó S. (2006). User's guide to sigf: Significance testing by approximate randomisation. URL: https://nlpado.de/~sebastian/software/sigf.shtml.

Pedregosa F. et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830. URL: https://jmlr.org/papers/v12/pedregosa11a.html.

References IV

Powers D. M. W. (2008). Evaluation Evaluation. *18th European Conference on Artificial Intelligence, Proceedings*. ECAI 2008. Patras, Greece: IOS Press, pp. 843–844. DOI: 10.3233/978-1-58603-891-5-843.

Rand W. M. (1971). Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850. DOI: 10.1080/01621459.1971.10482356.

Reimers N. and Gurevych I. (2017). Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. EMNLP 2017. Copenhagen, Denmark: Association for Computational Linguistics, pp. 338–348. DOI: 10.18653/v1/D17-1035.

Ribeiro M. T. et al. (2020). Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. ACL 2020. Online: Association for Computational Linguistics, pp. 4902–4912. DOI: 10.18653/v1/2020.acl-main.442.

van Rijsbergen C. J. (1979). Information Retrieval. 2nd Edition. London, UK: Butterworth-Heinemann. ISBN: 978-0-408-70929-3. URL: http://www.dcs.gla.ac.uk/Keith/Preface.html.

Saito T. and Rehmsmeier M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE*, vol. 10, no. 3, pp. 1–21. DOI: 10.1371/journal.pone.0118432.

Sakai T. (2006). Evaluating Evaluation Metrics Based on the Bootstrap. *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '06. Seattle, WA, USA: Association for Computing Machinery, pp. 525–532. DOI: 10.1145/1148170.1148261.

Sculley D. et al. (2015). Hidden Technical Debt in Machine Learning Systems. *Advances in Neural Information Processing Systems 28*. NIPS 2015. Montréal, QC, Canada: Curran Associates, Inc., pp. 2503–2511. URL: https://proceedings.nips.cc/paper/2015/file/86df7dcfd896fcaf2674f757a2463eba-Paper.pdf.

Segalovich I. (2010). Machine Learning in Search Quality at Yandex. Keynote Presentation at the Industry Track of the 33rd Annual ACM SIGIR Conference. URL: https://www.eurospider.com/images/SIGIR_2010/04_SIGIR-2010-SEGALOVICH.pdf.

Smucker M. D., Allan J., and Carterette B. (2007). A Comparison of Statistical Significance Tests for Information Retrieval Evaluation. *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*. CIKM '07. Lisbon, Portugal: Association for Computing Machinery, pp. 623–632. DOI: 10.1145/1321440.1321528.

Ustalov D. et al. (2019). Watset: Local-Global Graph Clustering with Applications in Sense and Frame Induction. *Computational Linguistics*, vol. 45, no. 3, pp. 423–479. DOI: 10.1162/COLI_a_00354.

Velardi P., Faralli S., and Navigli R. (2013). OntoLearn Reloaded: A Graph-Based Algorithm for Taxonomy Induction. *Computational Linguistics*, vol. 39, no. 3, pp. 665–707. DOI: 10.1162/COLI_a_00146. the content is released under a CC BY-NC-ND license, but used with the permission of The MIT Press.

References V

Voorhees E. M. (1999). The TREC-8 Question Answering Track Report. *Proceedings of the 8th Text REtrieval Conference*. TREC-8. Gaithersburg, MD, USA: NIST, pp. 77–82. URL: https://trec.nist.gov/pubs/trec8/papers/qa_report.pdf.

Wang Y. et al. (2013). A Theoretical Analysis of NDCG Type Ranking Measures. *Proceedings of the 26th Annual Conference on Learning Theory*. Vol. 30. Proceedings of Machine Learning Research. Princeton, NJ, USA: PMLR, pp. 25–54. URL: https://proceedings.mlr.press/v30/Wang13.html.

Yang J. and Leskovec J. (2013). Overlapping Community Detection at Scale: A Nonnegative Matrix Factorization Approach. *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*. WSDM '13. Rome, Italy: Association for Computing Machinery, pp. 587–596. DOI: 10.1145/2433396.2433471.

Yeh A. (2000). More accurate tests for the statistical significance of result differences. *Proceedings of the 18th Conference on Computational Linguistics - Volume 2*. COLING '00. Saarbrücken, Germany: Association for Computational Linguistics, pp. 947–953. DOI: 10.3115/992730.992783.

Supplementary Media I

Alexas_Fotos (October 7, 2017). Calculating Machine Resulta Old. Pixabay. URL: https://pixabay.com/images/id-2825179/. Licensed under Pixabay License.

Amos E. (December 19, 2011). The Vectrex video game console, shown with controller. Wikimedia Commons. URL: https://commons.wikimedia.org/wiki/File:Vectrex-Console-Set.jpg. Licensed under CC BY-SA 3.0, used with author's permission.

bamenny (February 24, 2016). Robot Flower Technology. Pixabay. URL: https://pixabay.com/images/id-1214536/. Licensed under Pixabay License.

Buissinne S. (August 25, 2016). Dictionary Reference Book Learning. Pixabay. URL: https://pixabay.com/images/id-1619740/. Licensed under Pixabay License.

Dumlao N. (November 21, 2017). two person pouring coffee with piled cups photo. Unsplash. URL: https://unsplash.com/photos/eksqjXTLpak. Licensed under Unsplash License.

Finnsson I. (May 19, 2017). Books Covers Book Case. Pixabay. URL: https://pixabay.com/images/id-2321934/. Licensed under Pixabay License.

Free-Photos (August 9, 2016). Person Mountain Top Achieve. Pixabay. URL: https://pixabay.com/images/id-1245959/. Licensed under Pixabay License.

Merrill B. (July 24, 2014). Pedestrians People Busy. Pixabay. URL: https://pixabay.com/images/id-400811/. Licensed under Pixabay License.

Nichtich (February 3, 2008). precision and recall in binary classification. Minor modifications: renamed P to Pr, R to Re, and added labels for TP, TN, FP, FN. Wikimedia Commons. URL: https://commons.wikimedia.org/wiki/File:Recall-precision.svg. Licensed under Public Domain.

Pexels (November 23, 2016). Aquarium Jellyfish Aquatic. Pixabay. URL: https://pixabay.com/images/id-1851643/. Licensed under Pixabay License.

Rahman Rony M. (May 31, 2016). Mad Max Fury Car Monster. Pixabay. URL: https://pixabay.com/images/id-1426796/. Licensed under Pixabay License.

rawpixel (April 18, 2017). Calm Freedom Location. Pixabay. URL: https://pixabay.com/images/id-2218409/. Licensed under Pixabay License.

rawpixel (June 23, 2018). Agreement Business Businessman. Pixabay. URL: https://pixabay.com/images/id-3489902/. Licensed under Pixabay License.

shbs (October 1, 2017). Petersburg Architecture Saint. Pixabay. URL: https://pixabay.com/images/id-2805503/. Licensed under Pixabay License.