A simple but thorough primer on metrics for multi-class evaluation such as Micro F1, Macro F1, Kappa and MCC

Juri Opitz
Heidelberg University

Different metrics are used to evaluate classifiers. Many recent papers and shared tasks pick 'macro' metrics to rank systems (e.g., 'macro F1'). However, sometimes the motivation for selecting a specific metric tends to be not fully clear, and the expectations associated with phrases like 'macro' seem blurry. Starting from the basic concepts of bias and prevalence, we analyze properties of metrics, with the aim of better metric understanding. In particular, we study Accuracy, macro Precision, macro Recall, macro F1, Matthews Correlation Coefficient, and Kappa.

1 Introduction

Consider a typical scenario in machine learning: We train a classifier to predict some categories of interest and want to assess its capability to predict unseen data.

To this aim, we usually evaluate the classifier's predictions against reference labels in two steps: First, we summarize the classifier's behavior in a *confusion matrix* that has a designated dimension for every possible prediction-label combination. Second, an aggregate statistic, which we here denote as *metric*, maps the confusion matrix onto a single number.

Obviously, a 'perfect' metric often doesn't exist, since we lose important information about a classifier's behavior when reducing the confusion matrix to a single number. Nonetheless, for classifier selection or ranking, a suitable metric has to be chosen.

Over the recent years, there has been a surge of papers that use 'macro' metrics for evaluation. In particular, 'macro F1' has become popular for comparing classifiers and determining shared task winners – its increasing popularity is also reflected in the enormous Google-books corpus (Figure 1).

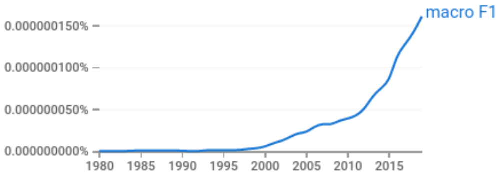


Figure 1: *Google books corpus n-gram search for macro F1.*

	metric	motivation
[1]	macro Prec, Recall, F1	'macro because (...) skewed distribution of the label set'
[13]	macro F1	'macro-averaging (...) implies that all class labels have equal weight in the final score'
[17]	macro F1	'Given the strong imbalance between the number of instances in the different classes'
[4]	accuracy, macro F1	'the labels are imbalanced'
[5, 15]	Mat. Corr. Coef. (MCC)	'balanced measurement when the classes are of very different sizes'
[14]	MCC, F1	'(...) imbalanced data (...)'

Table 1: *Example comments on evaluation metric selection.*

When searching for reasons why a particular metric has been selected, we tend to find rather unclear statements, e.g.: 'labels are imbalanced' or it is wished for that 'all class labels have equal weight' (c.f. Table 1).

But such statements seem to point at a certain user need: the evaluation score should not marginalize classification performance of data from infrequent classes. Instead, it should tell us a *bigger picture* of classifier capability (Greek: makrós, 'long') that is robust against fluctuating class prevalences. By contrast, a micro picture (Greek: mikrós, 'small') strictly binds the metric score to the class occurrences in a specific sample.

Outline After introducing *Preliminaries* (§2), we consider five *Metric Properties* (§3) to distinguish classification metrics. Then, we use these properties in an *Overview* (§4) of popular metrics, to improve our understanding about the metrics. Furthermore, we provide a *Discussion* (§5) that contains some considerations, and a short *Summary* (§6) of our notes.

2 Preliminaries

Basis For any classifier $f: D \rightarrow C = \{1, ..., n\}$ and finite set $S \subseteq D \times C$, let $m^{f,S} \in \mathbb{R}^{n \times n}$ be a confusion matrix where $m^{f,S}_{ij} = |\{s \in S \mid f(s_1) = i \wedge s_2 = j\}|$.[1] We omit superscripts whenever possible. A $metric: \mathbb{R}^{n \times n} \rightarrow \mathbb{R}$ allows us to order confusion matrices, respectively, rank classifiers. For convenience, we say that a classifier f is better than (or preferable to) a classifier g iff $metric(m^{f,S}) > metric(m^{g,S})$.

Let us define some basic quantities.

Class *bias, prevalence* and *correct* are given as

$$bias(i) = \sum_x m_{i,x} \qquad prevalence(i) = \sum_x m_{x,i}$$
$$correct(i) = m_{i,i}$$

Class precision P_i denotes the precision for class i:

$$P_i = \frac{correct(i)}{bias(i)} \approx \mathcal{P}(class = i | f \rightarrow i) \tag{1}$$

It approximates the probability of a correct prediction given that the classifier has predicted a specific class (which, for brevity, we denote as $f \rightarrow i$).

Class recall R_i denotes the recall for class i:

$$R_i = \frac{correct(i)}{prevalence(i)} \approx \mathcal{P}(f \rightarrow i | class = i) \tag{2}$$

It approximates the probability of a correct prediction given that an example is from a certain class.

3 Defining metric properties

We introduce some useful tools that help us to distinguish among metrics. We define five metric properties: *Monotonicity*, *class sensitivity*, *class decomposability*, *prevalence invariance* and *chance correction*.

I Monotonicity

This property checks the safety of a metric by imposing that correct (false) predictions do not decrease (increase) the score. We formalize this as

Property I (Monotonicity). *A $metric$ has PI if $\forall m$:*

$$\frac{\partial metric(m)}{\partial m_{i,j}} \begin{cases} \geq 0 \iff i = j \\ \leq 0 \iff i \neq j; \end{cases} \tag{3}$$

II Telling 'macro' from 'micro': different errors have different weights

Where there is 'macro', there is also 'micro'. We separate micro metrics with

Property II (Class sensitivity). *If $\exists m \in \mathbb{R}^{|C| \times |C|}_{\geq 0}$: $\frac{\partial metric(m)}{\partial m_{i,i}} \neq \frac{\partial metric(m)}{\partial m_{j,j}}$ with $(i,j) \in (C \times C)$ or $\frac{\partial metric(m)}{\partial m_{i,j}} \neq \frac{\partial metric(m)}{\partial m_{k,l}}$ with $(i,j,k,l) \in (C \times C \times C \times C)$ and $i \neq j, k \neq l$, then $metric$ is not a micro metric.*

A 'macro' metric, in contrast to a 'micro' metric, is sensitive to class labels.

III Macro average: It's a mean over classes

'Macro' metrics are sometimes named 'macro-average' metrics, which indicates that they may be perceived as an average over classes. We express this as

Property III (Class decomposability). *A 'macro-average' metric can be stated as*

$$metric(m, g, p) = \left(\frac{1}{n} \sum_{i=1}^{n} g(m_i, (m^T)_i)^p \right)^{\frac{1}{p}}. \tag{4}$$

I.e., if our $metric$ can be defined as an unweighted generalized mean over classes using a 'local' metric g that makes use of all instances related to a specific class ($class = i$ or $f \rightarrow i$), then we say that it has PIII.

IV More strict: "Treat all classes equally"

A common argument for using metrics other than the rate of correct predictions is that a metric should show '*classifier performance equally w.r.t. all classes*', or '*does not neglect rare classes*' (e.g., see [16, 7, 2, 13]).

Such a metric promises better extrapolation of the measured performance to data sets with different class frequencies. At first glance, PIII seems to imply such a feature already, since we compute an *unweighted mean* over classes. However, the score w.r.t. one class is (potentially) coupled to the prevalence of other classes.

So it makes sense to define such an expectation ('treat all classes equally') most strictly. We simulate different class prevalences with a

Prevalence scaling. We can use a diagonal prevalence scaling matrix λ to set

$$m' = m\lambda. \tag{5}$$

By scaling a column i with λ_{ii}, we inflate (or deflate) the mass of data that belong to class i (e.g., see Tables 2, 3, 4), but retain the relative proportions of intra-class error types. Now, we can define

[1] If our classifier predicts a n-dimensional probability distribution: $\sigma: D \rightarrow \{z \in [0,1]^n \mid \sum_{i=1}^{n} z_i = 1\}$, we may also set $m^{f,S}_{ij} = \sum_{s \in S} \mathbb{I}[s_2 = j] \cdot f(s_1)_i$, with $\mathbb{I}[x] = 1$ if x is true, else 0.

c f	a	b
a	15	5
b	10	10

Table 2: *b occurs 15 times.*

c f	a	b
a	$15 \cdot 1$	$5 \cdot 2$
b	$10 \cdot 1$	$10 \cdot 2$

Table 3: *Apply* $\lambda = (1, 2)$.

c f	a	b
a	15	10
b	10	20

Table 4: *b occurs 30 times.*

Property IV (Prevalence invariance). *If* $(\lambda, \lambda') \in \mathbb{R}^{n\times n}_{>0} \times \mathbb{R}^{n\times n}_{>0}$ *is a pair of diagonal matrices then* $metric(m\lambda) = metric(m\lambda')$.

PIV makes a metric invariant to class prevalence.

Prevalence calibration There is an interesting special case of λ. We can select λ s.t. all classes have the same prevalence. We call this prevalence calibration:

$$\lambda_{ii} = \frac{1}{n \cdot prevalence(i)}, \tag{6}$$

and achieve equal class prevalence (without changing the proportions of intra-class error types).

V Chance correction

There are two well-known chance 'baseline' classifiers: One predicts labels uniformly at random, the other based on prevalence. More arbitrary chance classifiers exist. The interpretability of a macro metric score increases if the score of the best chance classifier is not more than a function of the number of classes n. This indicates robustness against chance predictions and, in addition, makes the measurement of a single classifier meaningful by providing a chance baseline score.

Property V (Chance Correction). *A* $metric$ *has this property, if, for a dataset* S *with* n *classes and a set* A *that contains arbitrary random classifiers:*

$$\max\left\{metric(m^{r,S}) \mid r \in A\right\} = c(n),$$

where function c provides a score that is based on the number of classes n alone. In the special case of $\max\{metric(m^{r,S});\ r \in A\} = \min\{metric(m^{r,S});\ r \in A\}$, we say that $metric$ is *strictly chance corrected*, and in the special case that the latter holds for all data set pairs S, S', we say that $metric$ is *completely chance corrected*. Note that strictness or completeness isn't always desired, since it does not correct for accuracy.

4 Metrics: overview and analysis

Accuracy (aka Micro Recall aka Micro Precision aka Micro F1)

Before studying other metrics, we briefly view $accuracy$ that tells us the ratio of correct predictions:

$$accuracy = \frac{\sum_i m_{i,i}}{\sum_{(i,j)} m_{i,j}} = \frac{\sum_i correct(i)}{\sum_i prevalence(i)}. \tag{7}$$

It is equivalent to '*micro Prec., micro Recall and micro F1*' that are often shown in papers (c.f. Appendix A).

Property analysis and discussion We easily see that it has only PI (monotonicity). This is expected, since PII-V tend to target *macro* metrics.

In general, accuracy is a key evaluation statistic, approximating the probability to observe a correct prediction. However, clearly the metric is strictly tied to the class prevalences that occur in a specific data set, and researchers seem interested in other metrics.

Macro recall: ticks all boxes

Macro recall is calculated as the unweighted arithmetic mean over all class-wise recall scores:

$$macR = \frac{1}{n}\sum_i R_i \tag{8}$$

Property analysis Macro Recall has all five properties (Proofs in Appendix B). We also note that it is *strictly* chance corrected with $c(n) = 1/n$.

Discussion Since macro Recall has all five properties, it is a useful macro metric for transparent and meaningful classifier evaluation. Moreover, it offers three intuitive interpretations: *Drawing items from class bags*, *Bookmaker metric* and *prevalence-calibrated accuracy*.

In the first interpretation, we are given n random items, one from every class. We select a random one and ask: what's the probability that it is correctly predicted? $MacR$ knows the answer.

Macro Recall also has another interesting interpretation, since it can be viewed as **a (fair) Bookmaker's**

metric.[2] For every prediction, we pay 1 unit and gain units according to fair (European) odds. The odds for making a correct prediction, when the true class is i, are $odds(i) = \frac{|S|}{prevalence(i)}$. For each data example, our classifier f makes a bet, incurring a total

$$\begin{aligned}
gain &= \sum_{s \in S}\Big(\mathbb{I}[f(s_1) = s_2] \cdot odds(s_2) - 1\Big) && (9)\\
&= -|S| + \sum_{s \in S}\Big(\mathbb{I}[f(s_1) = s_2] \cdot odds(s_2)\Big) && (10)\\
&= -|S| + \sum_{i=1}^{n}\Big[odds(i) \cdot \sum_{\substack{s \in S\\ s_2 = i}}\Big(\mathbb{I}[f(s_1) = i]\Big)\Big] && (11)\\
&= -|S| + |S| \sum_{i=1}^{n} \frac{correct(i)}{prevalence(i)} && (12)\\
&= -|S| + n|S| \cdot macR && (13)
\end{aligned}$$

Ignoring the cost and normalizing by the number of classes and data size ($n|S|$), we obtain macro Recall.

Finally, we can view **macro Recall as an accuracy score after class prevalence calibration**. To see this, first consider the standard accuracy measure (Eq. 7) and calibrate class prevalence (Eq. 6), observing that

$$\begin{aligned}
macAcc &= accuracy(m\lambda) && (14)\\
&= \frac{\sum_i \lambda_{ii} \cdot correct(i)}{\sum_i \lambda_{ii} \cdot prevalence(i)} && (15)\\
&= \sum_i \lambda_{ii} \cdot correct(i) = macR. && (16)
\end{aligned}$$

Macro precision: In the bias lies the issue

Macro-precision is the unweighted arithmetic mean over the class-precision scores:

$$macP = \frac{1}{n}\sum_i P_i \qquad (17)$$

Property analysis While properties I, II, III, V are fulfilled, macro Precision does not have prevalence invariance (Proofs in Appendix C). We can find theoretic m, λ, where the maximum score difference ($macP(m)$ vs. $macP(m\lambda)$) approaches $1-\frac{1}{n}$. Same as macro Recall, it is strictly chance corrected with $c(n) = 1/n$

Discussion Precision of class i approximates $\mathcal{P}(class = i | f \rightarrow i)$. Hence, at a glance, $macP$ provides us with an overall measure of 'prediction trustworthiness', which may be valuable.

However, the issue is that $bias(i)$ is a function of the nature of a classifier *and* the data $prevalence(j), j \in C$. Therefore, even though it is decomposed over classes (PIII), it is not invariant to prevalence changes (PIV). If we have f, f' with different biases, score differences are difficult to interpret.

To mitigate this issue, we can make *classifier bias* more meaningful, by calibrating class prevalence (Eq. 6), controlling prevalence and (consequently) bias.

Macro F1: Metric of choice in many tasks

Macro F1 is often used for classifier evaluation.[3] It is defined as an unweighted arithmetic mean over class-wise harmonic means of precision and recall:

$$macF1 = \frac{1}{n}\sum_i F1_i = \frac{1}{n}\sum_i \frac{2P_iR_i}{P_i + R_i}. \qquad (18)$$

with $P_i, R_i, F_i = 0$ if the denominator is zero.

Property analysis Four properties are fulfilled (PI, PII, PIII, PV) while PIV is not fulfilled (Proofs in Appendix D). Interestingly, while macro F1 has PV (chance correction), the chance correction isn't strict, differentiating it from most other macro metrics. In particular, its chance baseline upper-bound $1/n$ is achieved when $\mathcal{P}(f \rightarrow i) = \mathcal{P}(class = i)$, which means that macro F1 not only corrects for chance, but also for accuracy.

Additionally, we see that macro F1 is invariant to the false-positive and false-negative error spread for a specific class. This can be seen by writing:

$$macF1 = \frac{2}{n}\sum_i \frac{correct(i)}{bias(i) + prevalence(i)}, \qquad (19)$$

and viewing the denominator.

Discussion While the invariance to error types and the targeted balance between chance correction and accuracy seem useful, macro F1 inherits an issue of macro Precision: *classifier bias* to a class i is tied to class prevalence of all j, $j \neq i$. Thus, in a strict sense, macro F1 does not 'treat all classes equally'. The score may become more meaningful after class prevalence calibration (Eq. 6).

Macro F1: a doppelganger

Interestingly, there is another frequently used metric that has been coined 'macro F1' [12]. It is the harmonic mean of macro Precision and macro Recall:

$$macF1' = \frac{2 \cdot macR \cdot macP}{macR + macP}. \qquad (20)$$

[2] For other bookmaker inspired metrics see [9, 10]

[3] It is also set as default in evaluation reports of popular machine learning packages such as `scikit-learn`.

Property analysis In contrast to its name twin, one less property is fulfilled (PIII), since it cannot be decomposed over classes (Proofs in Appendix E), and it is *strictly* chance baseline corrected with $c(n) = 1/n$. In [8], we provide more analyses of the two macro F1s.

Discussion On one hand, $macF1'$ isn't easy to interpret, since the numerator contains the cross-product of all class-wise recall and precision values. However, it can be viewed through the lens of an inter-annotator agreement (IAA) metric, where we do not compare a classifier against a reference, but instead we compare two reference candidates A and B, which results in confusion matrices $m_A = m$ and $m_B = m^T$. Therefore:

$$macF1' = \frac{2 \cdot macR(m_A) \cdot macR(m_B)}{macR(m_A) + macR(m_B)}, \tag{21}$$

falling back on $macR$'s clear interpretation(s).

Birds of a feather: Kappa and Matthews Correlation Coefficient

If we assume normalized confusion matrices[4], we can state both metrics as concise as possible. First denote

$$c = \sum_i correct(i) \qquad \mathbf{b} = m\mathbf{1} \qquad \mathbf{p} = m^T\mathbf{1}, \tag{22}$$

where c is the amount of correct predictions, $\mathbf{p}$ is a vector where at each index i we find $prevalence(i)$, and $\mathbf{b}$ is a vector where at each index i we find $bias(i)$.

Generalized Matthews correlation coefficient (MCC) The multi-class generalization of MCC [6] can now be written concisely as

$$MCC = \frac{c - \mathbf{p}^T\mathbf{b}}{(\sqrt{1 - \mathbf{b}^T\mathbf{b}})(\sqrt{1 - \mathbf{p}^T\mathbf{p}})}. \tag{23}$$

Cohen's kappa Let us state Cohen's kappa [3] as follows, to illuminate its similarity to MCC:

$$KAPPA = \frac{c - \mathbf{p}^T\mathbf{b}}{1 - \mathbf{p}^T\mathbf{b}}. \tag{24}$$

Property analysis MCC and KAPPA have PII and PV (complete chance baseline correction: $c(n) = 0$). Interestingly, they are *non*-monotonic metrics (PI). They also do not have PIII, PIV. Proofs are in Appendix F. Note also that MCC ≥ KAPPA (since $\mathbf{p}^T\mathbf{b} \leq \mathbf{b}^T\mathbf{b}, \mathbf{p}^T\mathbf{p}$).

Discussion Kappa and MCC are similar measures. Since $\mathbf{p}^T\mathbf{b} \approx \sum_i \mathcal{P}(c = i) \cdot \mathcal{P}(f \to i)$ is the probability of a random baseline classifier to correctly predict an item from a random class, and $c \approx \mathcal{P}(correct)$, KAPPA and MCC can be viewed as accuracy calibrated against a random chance baseline classifier.

However, overall they are calibrated in slightly different ways due to different denominators. The denominator of KAPPA simply shows the upper-bound w.r.t. the score of the perfect classifier, which is intuitive.

On the other hand, the denominator in MCC is harder to interpret due to its stronger dependence on *classifier bias*. This becomes evident when viewing the measures after class-prevalence calibration (Eq. 6):

$$\begin{aligned} KAPPA(m\lambda) &= \frac{c - n^{-1}}{1 - n^{-1}} \\ MCC(m\lambda) &= \frac{c - n^{-1}}{(\sqrt{1 - \mathbf{b}^T\mathbf{b}})(\sqrt{1 - n^{-1}})}, \end{aligned}$$

The influence of classifier biases can make it more difficult to meaningfully compare classifiers with MCC. By contrast, in KAPPA, the bias term is gone.

5 Discussion

Prevalence calibration of macro metrics

Macro Recall shows all five properties PI-V, and therefore offers good interpretability. However, researchers are also interested in using other macro metrics, such as macro Precision, or macro F1, to, e.g., target a $bias$ vs. $prevalence$ balance over all classes (Eq. 19). To make the results of these metrics more meaningful, we can consider calibrating class prevalence such that all classes have the same prevalence (Eq. 5, Eq. 6). Such a calibration makes a classifier's *bias* (i.e., Precision, F1) more meaningful and equips every metric with PIV.

Note that some popular software packages, e.g., `scikit-learn`, offer a calibration option, when creating confusion matrices from references and predictions.[5]

More considerations

Other means in macro Recall We can use other p in the generalized mean (Eq. 4), besides $p = 1$ (arithmetic mean). E.g., we can use the geometric mean ($p \to 0$):

$$GmacR = GM(R_1, ..., R_n) = \sqrt[n]{R_1 \cdot ... \cdot R_n} \tag{25}$$

[4] $m_{ij} = \frac{1}{|S|}|\{s \in S \mid f(s_1) = i \wedge s_2 = j\}| \in [0,1], \sum_{(i,j)} m = 1$. This shows probabilities for error types and does not change the MCC or KAPPA score.

[5] `from sklearn.metrics import confusion_matrix;`
`m = confusion_matrix(refs, preds, normalize='True'))`

metric	PI (mono.)	PII (class sens.)	PIII (decompose)	PIV (prev. invar.)	PV (chance correct)
macro Recall	✓	✓	✓	✓	✓: $1/n$, strict
macro Prec.	✓	✓	✓	✗ (✓)	✓: $1/n$, strict
macro F1	✓	✓	✓	✗ (✓)	✓: $1/n$
macro F1'	✓	✓	✗	✗ (✓)	✓: $1/n$, strict
Kappa	✗	✓	✗	✗ (✓)	✓: 0, complete
MCC	✗	✓	✗	✗ (✓)	✓: 0, complete
Accuracy	✓	✗	✗	✗	✗
Macro Acc.			same as macro Recall (Eq. 14-16)		
Micro F1			same as accuracy (see Appendix A)		

Table 5: *Summary of evaluation metrics. (✓): a property is fulfilled after prevalence calibration.*

GmacR, same as *macR*, has all five properties (PI-PV). Given n random items, one from every class, *GmacR* approximates a (class-count normalized) probability that all are correctly predicted.

GmacR can be useful when it is $important$ to have good performance for *all* classes.

Presenting class-wise recall scores helps to extrapolate metrics for new data We can estimate precision in another data set (under unknown class-prevalence and without reference), based on class-wise recall. First, we state an estimate of the class distribution $P(class) \approx \widehat{P(class)}$ that can be expected. Then we estimate $P(f) \approx \widehat{P(f)}$, by running the classifier on some data. Finally:

$$P^{\star}(class = i | f \rightarrow i) = \frac{P(f \rightarrow i | class = i) \cdot P(class = i)}{P(f \rightarrow i)} \tag{26}$$

$$\approx \frac{R_i \cdot \widehat{P(class = i)}}{\widehat{P(f \rightarrow i)}} \tag{27}$$

From here, estimated scores of macro metrics, including MCC and KAPPA, easily follow. Note that it is not possible to approximate expected recall values on new data by providing precision values on old data (since these do not transfer). Hence, this corroborates the value of recall statistics.

How do metrics behave in a practical evaluation setting? So far, we mainly focused on theory. In an empirical setting, differences between most metrics are unlikely to be as extreme as in hypothetically constructed cases. However, when studying results of a shared task on document classification, we can observe important differences in the rankings of classifiers (c.f. Appendix G). This underlines the importance of understanding metrics, to inform metric selection and achieve meaningful classifier rankings.

6 Conclusion

Table 5 shows an overview of the visited metrics.

We make some observations: i) only macro Recall has all five properties and it is the only metric that shows class prevalence invariance (PIV), i.e., 'it treats all classes equally' (in a strict sense). However, through prevalence calibration, all metrics obtain PIV. ii) In contrast to other metrics, KAPPA and Matthews Correlation Coefficient do not have property PI. I.e., under some circumstances, errors can increase the score, possibly lowering interpretability. iii) all metrics except accuracy show chance baseline correction. Strict chance baseline correction isn't a feature of Macro F1, and complete chance baseline correction (class-count independent) is only achieved with MCC and KAPPA.

In general, we can conclude that macro Recall and accuracy well complement each other. Both have a clear interpretation, and relate to each other with a simple prevalence calibration. In particular, macro Recall can be understood as a prevalence-calibrated version of accuracy. On the other hand, macro F1 is interesting since it does not strictly correct for chance (as in macro Recall) but also factors in more accuracy. MCC and KAPPA are similar measures, where KAPPA tends to have more robustness against classifier biases.

References

[1] Francesco Barbieri, Jose Camacho-Collados, Francesco Ronzano, Luis Espinosa-Anke, Miguel Ballesteros, Valerio Basile, Viviana Patti, and Horacio Saggion. SemEval 2018 task 2: Multilingual emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 24–33, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[2] Fabricio Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. Detecting spammers on twitter. In *Collaboration, electronic mes-*

saging, anti-abuse and spam conference (CEAS), volume 6, page 12, 2010.

[3] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[4] Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. SemEval-2021 task 6: Detection of persuasion techniques in texts and images. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 70–98, Online, August 2021. Association for Computational Linguistics.

[5] Xiaoan Ding, Tianyu Liu, Baobao Chang, Zhifang Sui, and Kevin Gimpel. Discriminatively-Tuned Generative Classifiers for Robust Natural Language Inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8189–8202, Online, November 2020. Association for Computational Linguistics.

[6] Jan Gorodkin. Comparing two k-category assignments by a k-category correlation coefficient. *Computational biology and chemistry*, 28(5-6):367–374, 2004.

[7] Neel Kant, Raul Puri, Nikolai Yakovenko, and Bryan Catanzaro. Practical text classification with large pre-trained language models. *arXiv preprint arXiv:1812.01207*, 2018.

[8] Juri Opitz and Sebastian Burst. Macro f1 and macro f1. *arXiv preprint arXiv:1911.03347*, 2019.

[9] David MW Powers. Recall & precision versus the bookmaker. 2003.

[10] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.

[11] Sara Rosenthal, Noura Farra, and Preslav Nakov. Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 502–518, 2017.

[12] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4):427–437, 2009.

[13] Cynthia Van Hee, Els Lefever, and Véronique Hoste. SemEval-2018 task 3: Irony detection in English tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[14] Frank Xing, Lorenzo Malandri, Yue Zhang, and Erik Cambria. Financial sentiment analysis: An investigation into common mistakes and silver bullets. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 978–987, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[15] Sanghyun Yi, Rahul Goel, Chandra Khatri, Alessandra Cervone, Tagyoung Chung, Behnam Hedayatnia, Anu Venkatesh, Raefer Gabriel, and Dilek Hakkani-Tur. Towards coherent and engaging spoken dialog response generation using automatic conversation evaluators. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 65–75, Tokyo, Japan, October–November 2019. Association for Computational Linguistics.

[16] Quan Yuan, Gao Cong, and Nadia Magnenat Thalmann. Enhancing naive bayes with various smoothing methods for short text classification. In *Proceedings of the 21st International Conference on World Wide Web*, pages 645–646, 2012.

[17] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

Appendices

A Accuracy aka micro Precision/Recall/F1

Micro F1 is defined[6] as the harmonic mean (HM) of 'micro Precision' and 'micro Recall', where micro Precision is

$$\frac{\sum_x correct(x)}{\sum_x bias(x)} \tag{28}$$

and micro Recall is

$$\frac{\sum_x correct(x)}{\sum_x prevalence(x)}. \tag{29}$$

Now it suffices to see that $\sum_x prevalence(x) = \sum_x bias(x) = \sum_{(i,j)} m_{i,j}$, and $\sum_x correct(x) = \sum_i m_{i,i}$, and $HM(a,a) = a$.

[6] E.g., see [12]

B Macro Recall

B.1 Monotonicity ✓

If $i \neq j$: $\frac{\partial macR(m)}{\partial m_{i,j}} = -\frac{correct(j)}{n \cdot prevalence(j)^2} \leq 0$; else $\frac{\partial macR(m)}{\partial m_{i,i}} = \frac{prevalence(i) - correct(i)}{n \cdot prevalence(i)^2} \geq 0$

B.2 Class sensitivity ✓

Follows from above.

B.3 Class decomposability ✓

In Eq. 4 set $g(row, col) = \frac{row_1}{\sum_i col_i}$ and $p = 1$.

B.4 Prevalence invariance ✓

$R'_i = \frac{\lambda_{i,i} m_{i,i}}{\sum_j \lambda_{i,i} m_{j,i}} = \frac{\lambda_{i,i} m_{i,i}}{\lambda_{i,i} \sum_j m_{j,i}} = R_i.$

B.5 Chance correction ✓

Assume normalized class prevalences $p \in [0,1]^n$ s.t. $\sum_{i=1}^n z_i = 1\}$ and arbitrary random baseline $z \in [0,1]^n$ s.t. $\sum_{i=1}^n z_i = 1\}$:

$$MacR = \frac{1}{n}\sum_i R_i = \frac{1}{n}\sum_i \frac{p_i \cdot z_i}{\sum_j z_j \cdot p_i} = \frac{1}{n}\sum_i z_i = \frac{1}{n} \tag{30}$$

C Macro precision

C.1 Monotonicity ✓

If $i \neq j$: $\frac{\partial macP(m)}{\partial m_{i,j}} = -\frac{correct(i)}{n \cdot bias(i)^2} \leq 0$; else $\frac{\partial macP(m)}{\partial m_{i,i}} = \frac{bias(i) - correct(i)}{n \cdot bias(i)^2} \geq 0$

C.2 Class sensitivity ✓

Follows from above.

C.3 Class decomposability ✓

In Eq. 4 set $g(row, col) = \frac{row_1}{\sum_i row_i}$ and $p = 1$.

C.4 Prevalence invariance

A counter-example $P'_i = \frac{\lambda_{i,i} m_{i,i}}{\sum_j \lambda_{j,j} m_{i,j}} \neq P_i$ is easily found. E.g., in Table 2, 3, 4: $macP = 0.5\frac{3}{4} + 0.5\frac{1}{2} = \frac{5}{8} \neq macP' = 0.5\frac{3}{5} + 0.5\frac{2}{3} = \frac{19}{30}$.

C.5 chance correction ✓

Assume normalized class prevalences $p \in [0,1]^n$ s.t. $\sum_{i=1}^n z_i = 1\}$ and arbitrary random baseline $z \in [0,1]^n$ s.t. $\sum_{i=1}^n z_i = 1\}$:

$$MacP = \frac{1}{n}\sum_i R_i = \frac{1}{n}\sum_i \frac{p_i \cdot z_i}{\sum_j z_i \cdot p_j} = \frac{1}{n}\sum_i p_i = \frac{1}{n} \tag{31}$$

D Macro F1

D.1 Monotonicity ✓

Let $Z_i = bias(i) + prevalence(i)$. If $i \neq j$:

$$\frac{\partial macF1(m)}{\partial m_{i,j}} = -\frac{2 \cdot correct(i)}{nZ_i^2} - \frac{2 \cdot correct(j)}{nZ_j^2} \leq 0 \tag{32}$$

else:

$$\frac{\partial macF1(m)}{\partial m_{i,j}} = \frac{2}{nZ_i} - \frac{2 \cdot correct(i)}{nZ_i^2} \tag{33}$$

$$+ \frac{2}{nZ_j} - \frac{2 \cdot correct(j)}{nZ_j^2} \geq 0 \tag{34}$$

D.2 Class sensitivity ✓

Follows from above.

D.3 Class decomposability ✓

In Eq. 4 set $p = 1$, $g(row, col) = \frac{2row_1}{\sum_x row_x + col_x}$

D.4 Prevalence invariance ✓

By counter-example, similar to (C.4).

D.5 chance correction ✓

Assume normalized class prevalences $p \in [0,1]^n$ s.t. $\sum_{i=1}^n z_i = 1\}$ and arbitrary random baseline $z \in [0,1]^n$ s.t. $\sum_{i=1}^n z_i = 1\}$. We have

$$MacF1 = \frac{1}{n}\sum_i \frac{2 \cdot p_i \cdot z_i}{\sum_j z_i \cdot p_j + \sum_j z_j \cdot p_i} = \frac{1}{n}\sum_i \frac{2 \cdot p_i \cdot z_i}{p_i + z_i}. \tag{35}$$

We see that a maximum is attained when p = z, and that this maximum is $\frac{1}{n}$.

E Macro F1 (name twin)

E.1 Monotonicity ✓

We have $\frac{\partial macF1'(m)}{\partial m_{i,j}} = \frac{2x}{macR+macP}$ where $x =$

$$\left(\frac{\partial macR(m)}{\partial m_{i,j}} + \frac{\partial macP(m)}{\partial m_{i,j}}\right) \tag{36}$$

$$\cdot \Big(macR + macP - macP \cdot macR\Big) \tag{37}$$

Since $macR$ and $macP$ have monotonicity and (37) ≥ 0, macF1' also has monotonicity.

E.2 Label sensitivity ✓

Follows from above.

E.3 Class decomposability

Not possible.

E.4 Prevalence invariance

See $macP$.

E.5 chance correction ✓

Since $macF1'$ is the harmonic mean from (strictly chance corrected) macro precision and macro recall, we also have strictly chance correction with $\frac{1}{n}$.

F KAPPA and MCC

F.1 Monotonicity

Kappa Let us state

$$KAPPA = \frac{cs - \mathbf{p}^{\mathbf{T}}\mathbf{b}}{s^2 - \mathbf{p}^{\mathbf{T}}\mathbf{b}} = \frac{N_K}{D_K} \tag{38}$$

with $s = \sum_{(i,j)} m_{i,j}$. Other variables were introduced before (Eq. 22). Now, let $z_{ij} = bias(j) + prevalence(i)$.

Then, iff $i \neq j$:

$$\frac{\partial KAPPA}{\partial m_{i,j}} = \frac{(c - z_{ij})D_K^2 - (2s - z_{ij})N_K}{D_K^2} \tag{39}$$

c f	a	b	c
a	10	43	0
b	1	1	0
c	0	0	1

Table 6: *MCC = **0.0**.*

c f	a	b	c
a	10	43	0
b	1	1	0
c	0	**10**	1

Table 7: *MCC = **0.07**.*

MCC Let us state

$$MCC = \frac{cs - \mathbf{p}^{\mathbf{T}}\mathbf{b}}{\sqrt{(s^2 - \mathbf{p}^{\mathbf{T}}\mathbf{p})(s^2 - \mathbf{b}^{\mathbf{T}}\mathbf{b})}} = \frac{N_M}{D_M} \tag{40}$$

Let now $v_{ij} = \frac{\partial p^T p}{\partial m_{i,j}} = 2 \cdot prevalence(j)$ and $u_{ij} = \frac{\partial b^T b}{\partial m_{i,j}} = 2 \cdot bias(i)$.

Then, iff $i \neq j$:

$$\frac{\partial MCC}{\partial m_{i,j}} = \frac{1}{2D_M^3}\Big[D_M^2(c - z_{ij}) \tag{41}$$

$$- N_M(2s - v_{ij})\sqrt{s^2 - \mathbf{b}^{\mathbf{T}}\mathbf{b}} \tag{42}$$

$$- N_M(2s - u_{ij})\sqrt{s^2 - \mathbf{p}^{\mathbf{T}}\mathbf{p}}\Big] \tag{43}$$

It suffices now to see that there exist configurations of confusion matrices where N_K (KAPPA) or N_M (MCC) → 0, but not $(c - z_{i,j}) \cdot D_{M|K}^2 \rightarrow 0$. □

An example, where MCC increases, when we add more errors, is described below in Tables 6 and 7:

F.2 Class sensitivity ✓

Trivial

F.3 Class decomposability

Trivial

F.4 Prevalence invariance

Trivial.

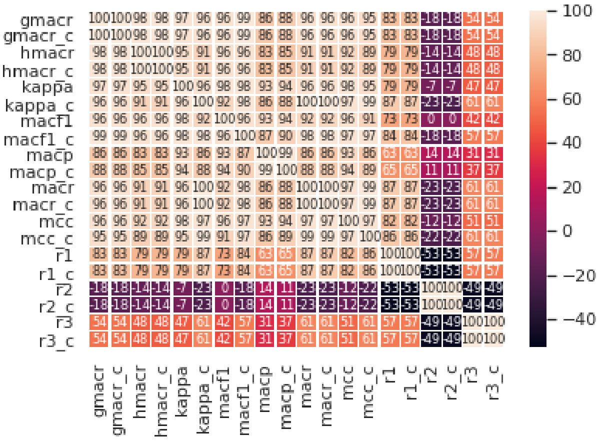
F.5 chance correction ✓

In the numerators we have

$$\sum_i p_i \cdot z_i - \sum_i \Big(\sum_j z_i \cdot p_j\Big)\Big(\sum_j z_j \cdot p_i\Big) = 0 \tag{44}$$

sys	macR	GmacR	HmacR	macF1	macF1c	kappa	mcc	r1	r2	r3
A	**68.1**	**66.8**	65.4	65.4	**67.7**	46.5	48.0	82.9	51.2	**70.2**
B	**68.1**	66.5	65.0	66.0	**67.7**	**47.3**	**49.2**	**87.8**	51.4	65.2
C	67.6	**66.8**	66.0	66.0	67.5	47.2	48.1	81.7	56.0	65.2
D	67.4	66.5	65.6	65.1	67.1	46.3	47.3	80.3	54.2	67.6
E	66.9	**66.8**	**66.8**	**66.0**	67.3	47.0	47.0	69.8	**64.0**	66.8
F	65.9	65.6	65.4	64.5	66.1	45.0	45.4	73.5	58.7	65.6
G	64.9	64.2	63.5	63.4	64.9	43.0	43.8	77.4	53.9	63.5
H	64.5	64.5	64.5	63.7	64.9	43.6	43.6	65.3	63.6	64.5

Table 8: *Shared task ranking with different metrics. r_i: recall for class i.*



	gmacr	gmacr_c	hmacr	hmacr_c	kappa	kappa_c	macf1	macf1_c	macp	macp_c	macr	macr_c	mcc	mcc_c	r1	r1_c	r2	r2_c	r3	r3_c
gmacr	100	100	98	98	97	96	96	99	86	88	96	96	96	95	83	83	-18	-18	54	54
gmacr_c	100	100	98	98	97	96	96	99	86	88	96	96	96	95	83	83	-18	-18	54	54
hmacr	98	98	100	100	95	91	96	96	83	85	91	91	92	89	79	79	-14	-14	48	48
hmacr_c	98	98	100	100	95	91	96	96	83	85	91	91	92	89	79	79	-14	-14	48	48
kappa	97	97	95	95	100	96	98	98	93	94	96	96	98	95	79	79	-7	-7	47	47
kappa_c	96	96	91	91	96	100	92	98	86	88	100	100	97	99	87	87	-23	-23	61	61
macf1	96	96	96	96	98	92	100	96	93	94	92	92	96	91	73	73	0	0	42	42
macf1_c	99	99	96	96	98	98	96	100	87	90	98	98	97	97	84	84	-18	-18	57	57
macp	86	86	83	83	93	86	93	87	100	99	86	86	93	86	63	63	14	14	31	31
macp_c	88	88	85	85	94	88	94	90	99	100	88	88	94	89	65	65	11	11	37	37
macr	96	96	91	91	96	100	92	98	86	88	100	100	97	99	87	87	-23	-23	61	61
macr_c	96	96	91	91	96	100	92	98	86	88	100	100	97	99	87	87	-23	-23	61	61
mcc	96	96	92	92	98	97	96	97	93	94	97	97	100	97	82	82	-12	-12	51	51
mcc_c	95	95	89	89	95	99	91	97	86	89	99	99	97	100	86	86	-22	-22	61	61
r1	83	83	79	79	79	87	73	84	63	65	87	87	82	86	100	100	-53	-53	57	57
r1_c	83	83	79	79	79	87	73	84	63	65	87	87	82	86	100	100	-53	-53	57	57
r2	-18	-18	-14	-14	-7	-23	0	-18	14	11	-23	-23	-12	-22	-53	-53	100	100	-49	-49
r2_c	-18	-18	-14	-14	-7	-23	0	-18	14	11	-23	-23	-12	-22	-53	-53	100	100	-49	-49
r3	54	54	48	48	47	61	42	57	31	37	61	61	51	61	57	57	-49	-49	100	100
r3_c	54	54	48	48	47	61	42	57	31	37	61	61	51	61	57	57	-49	-49	100	100

Figure 2: *Team ranking correlation matrix with paired metrics. metric_c means that the confusion matrix has been calibrated before metric computation (Eq. 6).*

G Small empirical study

We use metrics to rank teams that participated in the Semeval-2017 task [11] to predict the sentiment of tweet documents: negative, neutral, or positive.

The two winning systems were determined with macR, which is fair (A, B, Table 8). Yet, system E also does quite well, because it achieves a better balance over the three classes ($R_1 = 69.8, R_2 = 64.0, R_3 = 66.8$, max. $\Delta = 5.8$) as opposed to, e.g., *system B* ($R_1 = 87.8, R_2 = 51.4$, $R_3 = 65.2$ max. $\Delta = 36.4$), and indicated also by $GmacR$ and $HmacR$ metrics. Hence, if a user wants to ensure that all classes are well predicted under high uncertainty of prevalence (perhaps to be expected in Twitter data?), they may prefer system E.

Figure 2 shows a pair-wise Spearmanr correlation of team rankings. When considering the same metric before and after prevalence calibration, we see that only recall metrics agree with each other in their rankings. In particular, as indicated before, it seems that the second class is the one that may tip the scale: R_2 observably disagrees in its team ranking with *all* other metrics (Spearman's $\rho \leq 14$). Furthermore, $KAPPA$ assigns the exact ranks as $macR$, but only after prevalence calibration (Spearman's ρ Kappa$_c$ vs. macR: 100).