Better than Average: Paired Evaluation of NLP Systems

Maxime Peyrard[*]**, Wei Zhao**[†]**, Steffen Eger**[†]**, Robert West**[*]
[*]EPFL, Switzerland
[†]Technische Universität Darmstadt, Germany
{maxime.peyrard,robert.west}@epfl.ch
{zhao,eger}@aiphes.tu-darmstadt.de

Abstract

Evaluation in NLP is usually done by comparing the scores of competing systems independently averaged over a common set of test instances. In this work, we question the use of averages for aggregating evaluation scores into a final number used to decide which system is best, since the average, as well as alternatives such as the median, ignores the pairing arising from the fact that systems are evaluated on the same test instances. We illustrate the importance of taking the instance-level pairing of evaluation scores into account and demonstrate, both theoretically and empirically, the advantages of aggregation methods based on pairwise comparisons, such as the Bradley–Terry (BT) model, a mechanism based on the estimated probability that a given system scores better than another on the test set. By re-evaluating 296 real NLP evaluation setups across four tasks and 18 evaluation metrics, we show that the choice of aggregation mechanism matters and yields different conclusions as to which systems are state of the art in about 30% of the setups. To facilitate the adoption of pairwise evaluation, we release a practical tool for performing the full analysis of evaluation scores with the mean, median, BT, and two variants of BT (Elo and TrueSkill), alongside functionality for appropriate statistical testing.

1 Introduction

Research is driven by evaluation results, with attention and resources being focused on methods identified as state of the art (SotA). The proper design of evaluation methodology is thus crucial to ensure progress in the field. In NLP, evaluation usually consists in comparing the averaged scores of competing systems over a common set of test instances. Indeed, averaging scores independently for each system and declaring the one with the highest average to be best is particularly
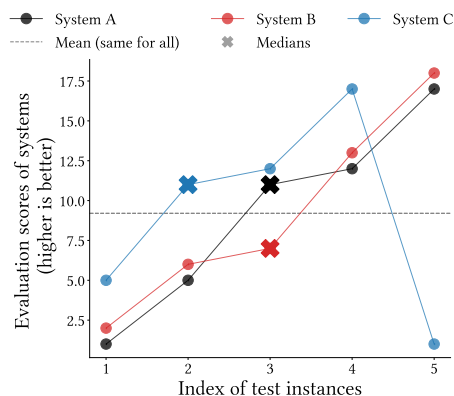

Figure 1: **Motivating example** (synthetic data). Evaluation scores of systems *A*, *B*, and *C* for five test instances. All systems have the same mean. *C* is better than *A* on all instances but one, so BT declares $C > A$ Also, *B* is better than *A* on all instances but one, so BT declares $B > A$, whereas the median of *A* is greater, and the means are the same. Overall, mean and median fail to capture the complex instance-level pairing.

simple, well understood, and mirrors the expected risk minimization paradigm used to train systems.

Here, we critically assess the specific choice of the average to aggregate evaluation scores. In particular, we emphasize that there is a natural *instance-level pairing* between the evaluation scores of systems, which aggregation mechanisms such as the mean or median fail to take into account: as they produce a score for each system independently, systems that have the same set of scores (but potentially in different order) cannot be distinguished.

Consider the three systems *A*, *B*, and *C* compared on five test instances in Fig. 1. Despite a complex pairing structure, they all have the same mean score across test instances. Moreover, even though *B* is better than *A* on all test instances but one, the median of *A* is greater than the median of *B*.

In this work, we discuss an alternative aggregation mechanism: the Bradley–Terry (BT) model

(Bradley and Terry, 1952). BT compares systems for each test instance and estimates the latent strength of systems based on how frequently one system scores higher than another. Such paired mechanisms have already been successfully used to aggregate human judgments (Novikova et al., 2018; Sedoc and Ungar, 2020); for example, WMT evaluation protocols regularly employ TrueSkill (Herbrich et al., 2007), a Bayesian variant of BT (Sakaguchi et al., 2014).

Contributions. We contribute the first comprehensive analysis of the BT model (especially vis-à-vis mean and median) as an aggregation mechanism for comparing system scores in NLP.

(i) We illustrate the importance of accounting for instance-level pairing and discuss the conditions under which the mean, median, and BT disagree about the ordering of systems. In Sec. 3, we draw parallels with the field of statistical testing, where *paired statistical tests* are recommended when comparing paired variables. Thus, we argue that paired aggregation mechanisms such as BT are more robust alternatives to the mean and median. We support this argument with simulations in Sec. 4.

(ii) We show that the differences between mean, median, and BT matter in practice. By re-evaluating 296 real NLP evaluation setups across four tasks and 18 evaluation metrics, different aggregation mechanisms yield different conclusions as to which systems are SotA in about 30% of the setups (Sec. 5). These results hold when replacing BT by the Elo (Elo, 1978) and TrueSkill variants.

(iii) We discuss further advantages and potential limitations of BT, alongside possible resolutions, in Sec. 7.

(iv) We recommend replacing the mean by BT in future evaluations of NLP systems. To ease the adoption of more robust aggregation mechanisms, we release *Pairformance*,[1] a practical tool for performing full analyses of evaluation scores with mean, median, BT, and two variants of BT (Elo and TrueSkill). The tool reports paired evaluation results alongside appropriate statistical testing for all five aggregation mechanisms and various visualization functionalities to elucidate the pairing structure between system scores.

Code and data for replicating our analyses and experiments is available online.[2]

---

[1] https://github.com/epfl-dlab/pairformance
[2] https://github.com/epfl-dlab/BT-eval

2 Aggregation of evaluation results

In this section, we briefly present the three aggregation mechanisms we consider.

2.1 Terminology

A standard evaluation setup typically consists of four elements:

1. At least two **systems,** A and B, to compare, with latent strengths λ_A and λ_B that we aim to estimate.

2. A **test set** $T = \{(x_l, y_l) : l = 1, \ldots, n\}$ consisting of n test instances, where x_l is the input and y_l is the ground-truth target output.

3. An **evaluation metric** M for scoring system outputs based on target outputs y_l, resulting in the sequence of **evaluation scores** $\mathcal{M}_A = \langle M(A(x_l), y_l) : l = 1, \ldots, n \rangle$ for system A.

4. An **aggregation mechanism** Θ that decides whether system A is better than B based on the evaluation scores of the two systems. We use $\Theta_{T,M}(A, B) = \Theta(\mathcal{M}_A, \mathcal{M}_B)$ to denote the comparison mechanism between A and B on the test set T with evaluation metric M. Here, Θ outputs its guess about which system is the best (or declares the comparison inconclusive if the difference is not statistically significant). For simplicity, we drop the dependency on T and M in the notation, simply writing $\Theta(A, B)$.

For example in text summarization, x_l is a source document from the test set, y_l its corresponding reference summary, and M might be ROUGE (Lin, 2004). The decision mechanism Θ usually compares the individual systems' mean evaluation scores, where the system with the highest mean score (here mean ROUGE score) is declared better.

Consistent evaluation result. We say that the outcome of such an evaluation is *consistent* if it recovers the ordering of systems implied by the inherent strengths of systems: $\Theta(A, B) = A \iff \lambda_A > \lambda_B$.

Probabilistic model. As commonly done in the literature on statistical testing, we view the evaluation scores of a system A as n indexed random variables: $X_A^{(l)}$, $l = 1, \ldots, n$, where n is the size of the test set. Note that this sequence of random variables is not necessarily i.i.d. Furthermore, even though systems A and B are independent, their evaluation scores are not, since there is an instance-level **pairing.** Intuitively, knowing the score of A on an instance (x_l, y_l) can provide information about the expected

2302

performance of B. For example, if A scores highly because (x_l, y_l) is an easy instance, one might expect B to also score highly.

2.2 Aggregation mechanisms

We now introduce three aggregation mechanisms Θ. We investigate their properties in subsequent sections.

Mean. This is the current standard: the system with the highest average score is declared the strongest. We denote this aggregation mechanism as MEAN. The average score of system A is computed as $E_A = \frac{1}{n} \sum_{l=1}^{n} X_A^{(l)}$.

Median. The median is an interesting alternative to the mean because it is robust to outliers. Here, the system with the highest median score is declared to be the strongest. The median score M_A of a system A is the central value in the sorted list of evaluation scores of A. We denote this aggregation mechanism as MEDIAN.

Bradley-Terry. The third option examined here is the Bradley–Terry (BT) model (Bradley and Terry, 1952). While MEAN and MEDIAN compute scores for systems A and B independently, BT is a function of the joint random variable $\left(X_A^{(l)}, X_B^{(l)} \right)$. BT estimates the relative strengths $\hat{\lambda}_A$ and $\hat{\lambda}_B$ of the two systems A and B, by comparing the evaluation scores for each test instance:

$$\mathbb{P}(A > B) = \frac{\hat{\lambda}_A}{\hat{\lambda}_A + \hat{\lambda}_B}. \qquad (1)$$

Intuitively, $\mathbb{P}(A > B)$ is the probability that, for any given test instance, A scores higher than B. The BT model chooses $\hat{\lambda}_A$ and $\hat{\lambda}_B$ in order to best explain the observations. The system with the highest $\hat{\lambda}$ is declared strongest.

When considering only two systems, the latent strength $\hat{\lambda}_A$ is the number of instances for which A scores better than B (and similarly for $\hat{\lambda}_B$). When the number of systems is greater than two, BT solves an iterative optimization algorithm that is guaranteed to converge to a unique solution (Bradley and Terry, 1952). We give details about BT and its computation in the general case in Appendix E.

We denote as BT the decision mechanism based on the BT model. While it is much less common than MEAN and MEDIAN, we will see below that BT satisfies interesting properties making it a more robust alternative.

3 Comparison of assumptions

Since the roles played by A and B are symmetrical, we now assume without loss of generality that system A is better, i.e., $\lambda_A > \lambda_B$.

Proposition 1. *If $\lambda_A > \lambda_B$ then*

- MEAN *consistent* $\iff E_A - E_B > 0$,
- MEDIAN *consistent* $\iff M_A - M_B > 0$,
- BT *consistent* $\iff M_{A-B} > 0$,

where E_S and M_S are the mean and median of the evaluation scores of system S, and M_{A-B} is the median of the differences between the evaluation scores of A and B. Note that $E_S, M_S,$ and M_{A-B} are all random variables.

The proof is given in Appendix B. Note that, whereas the expectation is linear ($E_A - E_B = E_{A-B}$), the median is not (in general, $M_A - M_B \neq M_{A-B}$).

Robustness to ouliers. The mean is not robust to outliers: E_{A-B} can be swayed above or below the threshold of 0 by a small number of test instances for which the difference between system scores is large. On the contrary, the median is a robust statistic that cannot be easily influenced by outliers. Similarly, BT is robust to outliers because its decision is based on the median of differences M_{A-B}.

Importance of pairing. The critical difference between BT, MEAN, and MEDIAN, is that only BT preserves the pairing information. Both MEAN and MEDIAN compute a statistic from the (unordered) set of scores $X_A^{(l)}$ and $X_B^{(l)}$ independently and then compare the aggregate statistics, losing the pairing structure. If the pairing actually does not matter, any permutation of the indices of system scores leaves the distribution of paired evaluation scores unchanged. This happens, for example, when both $X_A^{(l)}$ and $X_B^{(l)}$ are i.i.d.[3]

However, in the general case, the pairing matters. One particular example is when there exist different types of test instances and systems behave differently for different types, e.g., when there are *easy* instances on which all systems have higher scores. For example, consider the three systems and their evaluation scores on five test instances in Fig. 1. System A is worse than C on all instances but one, so $C > A$ according to BT, yet the median of A is greater than the median of C (10 vs. 7). At the same time, B outperforms C on all instances

---

[3]More generally, when the two sequences of random variables are exchangeable.

but one, so $B > C$ according to BT. For MEDIAN and MEAN, which ignore the pairing, A and B are completely equivalent, even though there is a clear difference regarding which system is more likely to be the best. This difference is revealed in the pairing structure. In general, any mechanism ignoring the pairing cannot capture the difference between A and B.

Choosing an aggregation mechanism. In Prop. 1, we stated the conditions for each mechanism to be *consistent*. Choosing an aggregation mechanism for a specific evaluation setup boils down to deciding what condition is more likely to hold in the setup. Note that none of the conditions implies any other condition in Prop. 1.

When comparing BT against MEAN (or MEDIAN), there are three possible scenarios: (i) BT agrees with MEAN (or MEDIAN), (ii) BT is consistent but MEAN (or MEDIAN) is not, and (iii) MEAN (or MEDIAN) is consistent but BT is not.

In case (i), it does not matter whether we use BT or MEAN (or MEDIAN).

In case (ii), for most instances, the better system has a higher score than the worse system, but MEAN (or MEDIAN) fails. For example, MEAN may be swayed by outliers, and MEDIAN may be swayed by jumps in score lists as in the example above.

In case (iii), for most instances, the better system has a lower score than the worse system, yet particular variations in the marginals make the MEAN or MEDIAN get the ordering correct. This is a very peculiar scenario: for MEAN, it implies that on the few instances on which the better system did better, the difference between evaluation scores was large enough to lift the mean of the better system above the other. We argue that if one really believes that the evaluation setup is likely to be in case (iii), then one does not trust the evaluation setup in the first place. It corresponds to assuming that the observed scores are inconsistent for the majority of test instances. If this is the case, one should rather improve the evaluation setup (e.g., metric, test set) in order to be more representative of the phenomena that one desires to capture.

Overall, the condition making BT consistent appears to be the most natural one. Trusting MEAN or MEDIAN more than BT implies holding an unintuitive belief about the evaluation setup, namely that the better system does worse than the worse system on a majority of test instances.

From another perspective, the random variables $E_A - E_B$ (MEAN) and $M_A - M_B$ (MEDIAN) are less likely to be (correctly) greater than zero in the presence of (i) complex pairing structures or (ii) outliers. The variable M_{A-B} (BT), on the contrary, is not affected by complex pairings or outliers.

3.1 Graphical criterion

Fig. 2 summarizes the problem of ignoring the pairing and offers a graphical criterion to understand the decisions made by MEAN, MEDIAN, and BT. In each plot, the densities are estimated by placing test instances at coordinates given by the evaluation scores of the two systems. The evaluation scores of A (green) are on the x-axis, and the evaluation scores of B (blue) on the y-axis. We also plot the *marginal distributions* of evaluation scores, from which we can read off means and medians. When the mean of $X_B^{(l)}$ is greater than that of $X_A^{(l)}$, the two extended lines representing the means meet in the upper triangle (above the line $X_A = X_B$), and analogously for the median. But mean and median being only functions of the marginals, they completely ignore the pairing. Fig. 2 illustrates this by depicting three completely different pairing structures where the marginals (and thus the means and medians) of A and B remain unchanged. (In Appendix A.1, we explain how to generate infinitely many such examples.) On the contrary, BT, being a property of the pairing (the 2D density), predicts that B is better than A when there is more mass in the upper triangle, i.e., more instances for which B scores higher than A. In the middle figure, the pairing indicates that A is better than B, in disagreement with the decisions of MEAN and MEDIAN.

3.2 Connection with statistical testing

The above discussion about the differences between MEAN, MEDIAN, and BT has interesting parallels with statistical testing.

When comparing the means of two systems over the same test set, the recommended statistical test is the *paired t*-test (Fisher, 1935). When comparing medians instead of means, the appropriate test is the sign test, which measures whether the median of the difference is significantly differerent from zero. Interestingly, the statistic of the sign test is precisely the one in the condition for BT to be consistent (see Prop. 1). Wilcoxon's signed-rank test (Wilcoxon, 1945) is often used as an alternative to the sign test because it has more statistical power (at the cost of making more assumptions). However,
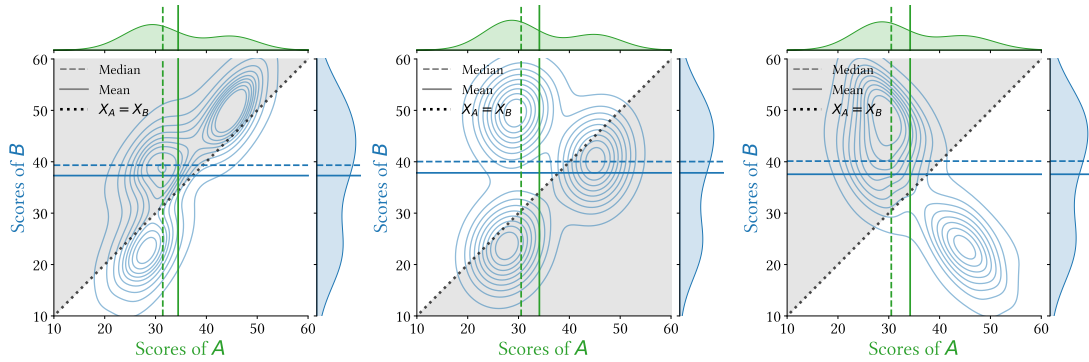
Figure 2: These 2D plots represent the distribution of test instances with coordinates given by the scores of the two systems being compared, i.e., the x-axis is the score $X_A^{(l)}$ of system A on some test instance (x_l, y_l), and the y-axis is the score $X_B^{(l)}$ of system B on the same instance. While the 3 plots represent different instance-level performances of A and B, the marginal (unpaired) distribution of scores of A and B remain unchanged. From such 2D plots, not only do we see the global structure of the pairing between the scores of A and B, we can also read off the decision of MEAN, MEDIAN and BT based on simple **geometrical criteria:** (i) if the prolongation of the means intersect above the $X_A = X_B$ line, then MEAN predicts that A is better, (ii) if the prolongation of the medians intersect above the $X_A = X_B$ line, then MEDIAN predicts that A is better, (iii) if there is more mass in the upper-left triangle, then BT predicts that system A is better. The latter case corresponds to most of the test instances being located in the upper-left triangle ($A > B$). The half-space with more mass is shaded.

Divine et al. (2018) showed that Wilcoxon's signed-rank test does not always properly account for the pairing of data, unlike the sign test.

When performing statistical testing, it seems obvious that we should use the paired version of tests when the data is naturally paired (Rankel et al., 2011). Even works discussing statistical testing in NLP recommend Wilcoxon's signed-rank test (Graham, 2015; Owczarzak et al., 2012; Dror et al., 2018). Yet, to obtain aggregated scores for systems, the community still mostly uses aggregation mechanisms that ignore the pairing, such as MEAN. MEDIAN is the outlier-resistant version of MEAN, and BT is the paired variant of MEDIAN. Whenever one recommends a paired test of medians, such as the sign test or Wilcoxon's signed-rank test, to obtain p-values, one should use BT to compare system scores.

4 Simulations with synthetic data

Next, we perform simulations to extend the analysis of the previous section to (i) $N > 2$ systems, (ii) finitely many test samples, (iii) a practical implementation of BT (for $N > 2$ systems, BT is an iterative optimization algorithm, as discussed in Appendix E).

We synthesize evaluation scores with various properties starting with systems of predefined implicit strengths λ_i. To create situations where the pairing of evaluation scores matters, we introduce multiple test instance types. For each type, systems perform differently but still have the same relative strength ($\mathbb{P}(A > B)$), differing only by an added offset. For example, the evaluation scores obtained by A and B could be sampled from $\mathcal{N}(\lambda_A, \sigma)$ and $\mathcal{N}(\lambda_B, \sigma)$ for one test instance type, and by $\mathcal{N}(\lambda_A + \epsilon, \sigma)$ and $\mathcal{N}(\lambda_B + \epsilon, \sigma)$ for another type, with ϵ being the offset. We sample evaluation setups by varying the following properties: the number of systems, the number of test instances, the percentage of outliers, the numbers of test instance types, and the level of noise. This results in 2,880 simulated evaluation setups. In Appendix A.2, we give the detailed algorithm and parameters used to generate the data.

In Fig. 3, we report Kendall's τ between the latent scores λ_i and the aggregated scores estimated by MEAN, MEDIAN, and BT. When the evaluation setup does not present any difficulty (Fig. 3(a, b)), all aggregation mechanisms work equally well (within each other's 95% error bounds), improving with more samples (Fig. 3(b)) and deteriorating with more systems (Fig. 3(a)). Unsurprisingly, MEAN fails in the presence of outliers, whereas MEDIAN and BT are unaffected (Fig. 3(c, e, f)). When several types of test instances are considered, MEDIAN begins to fail (Fig. 3(d)), which is made worse when outliers are also present (Fig. 3(f)). Overall, BT is more robust and does not fail when the pairing matters Fig. 3(g, h).
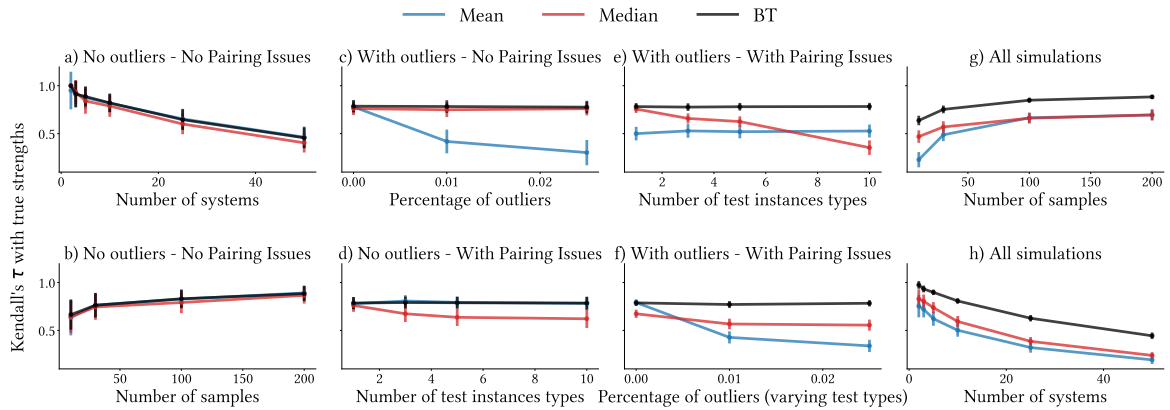
2305

Figure 3: The y-axis is the Kendall's τ correlation between latent scores λ_i of systems and the scores obtained after aggregating simulated evaluation scores with MEAN, MEDIAN, or BT. Fig. 3(a) and Fig. 3(b) corresponds to the intuitive case where no problem occurs (no outliers, no pairing issues). Fig. 3(c) adds outlier problems only, and Fig. 3(d) adds pairing issues only by increasing the number of types of test instances. Fig. 3(e) and (f) show the combined effect of outliers and pairing issues. Finally, Fig. 3(g) and Fig. 3(h) collect all the simulations. The error bars represent 95% confidence intervals obtained with bootstrap resampling.

5 Analysis of empirical data

In this section, we perform large-scale experiments using real evaluation scores from four NLG tasks. For summarization, we use the TAC-08, TAC-09, TAC-11 and CNN/DM (Hermann et al., 2015) datasets. For machine translation, we use the shared tasks of WMT-17 (Bojar et al., 2017), WMT-18 (Ma et al., 2018), and WMT-19 (Ma et al., 2019). For image captioning, we use the MSCOCO (Lin et al., 2014) dataset, and for dialogue, we use the PersonaChat and TopicalChat (Mehri and Eskenazi, 2020) datasets. The evaluation scores are obtained with a total of 18 different evaluation metrics: BLEU-[1,2,3,4] (Papineni et al., 2002), ROUGE-[1,2,L] (Lin, 2004), ROUGE-WE-[1,2] (Ng and Abrecht, 2015), JS-[1,2] (Lin et al., 2006), S3-[pyr, resp] (Peyrard et al., 2017), CIDEr (Vedantam et al., 2015), Chrfpp (Popovic, 2017), METEOR (Lavie and Agarwal, 2007), MoverScore (Zhao et al., 2019), and BERTScore (Zhang et al., 2020). Some metrics are only available for some task; e.g., CIDEr, METEOR are only available for the image captioning task. We provide details about datasets, metrics, and their statistics in Appendix A.3.

Overall, across datasets and metrics we have 296 evaluation setups, 73,471 pairs of systems, and 91,197 test instances. We also experiment with sub-sampling different sizes of test sets (see Appendix A.3) to simulate varying train/dev/test splits or cross-validation.

5.1 Comparison of BT, MEAN, and MEDIAN

In Table 1, we report the disagreement between aggregation mechanisms over all the data with three measures: the percentage of pairs ranked in a different order (rescaled version of Kendall's τ), the percentage of setups where the state-of-the-art (SotA) systems are different, and the percentage of setups where the top 3 systems are different (compared as sets). A significant fraction of pairs of systems (about 10%) are ranked differently by different mechanisms. More importantly, top systems are often different (in about 40% of setups for top 1 and 50% for top 3). We can conclude that the choice of aggregation mechanism has a real impact on evaluation outcome. The observed disagreement between the three aggregation metrics implies that we are not in the case depicted by Fig. 3(a) and Fig. 3(b), i.e., the pairing matters and there are outliers in real data. In the next paragraphs, we break down the disagreement per evaluation metric, task, and test set size. Detailed results are provided in Appendix C.

Which metrics are impacted most? We report in Fig. 4(a) the percentage of disagreement between aggregation mechanisms per metric averaged over datasets, when subsampling test sets of different sizes uniformly (see Appendix A.3 for details). While most metrics are available for all four tasks, METEOR and CIDEr are only available for the captioning task. Therefore, the observed disagreements for these metrics may be a feature of the task instead of the metrics. Interestingly, recent metrics

	Disagree	$\neq$ SotA	$\neq$ Top-3
MEAN VS. MEDIAN	4%	18%	30%
MEAN VS. BT	9%	40%	49%
MEDIAN VS. BT	9%	41%	55%

Table 1: Disagreement between aggregation mechanisms. The first column shows the percentage of system pairs ordered differently by two aggregation mechanisms. The second column shows the percentage of setups where two aggregation mechanisms find different SotA, and the third column shows the percentage of setups where the top-3 systems are different (compared as sets).

such as BERTScore and MOVERScore seem less affected. On the other hand, BLEU variants are the most impacted, particularly when comparing MEAN or MEDIAN against BT. The disagreement between MEAN and MEDIAN is stable across metrics. In general, MEAN and MEDIAN are more in agreement with one another than they are with BT, which indicates that pairing issues have a stronger effect than outliers.

Which tasks are impacted most? Fig. 4(b) summarizes an analysis as above, but across tasks instead of metrics. Again, to control for the fact that some tasks may have larger datasets, we subsample uniformly from various test set sizes. The results are averaged over evaluation metrics. Machine translation and summarization suffer the least while dialogue and image captioning display larger disagreement between aggregation mechanisms. This suggests important future research directions to improve the evaluation setups in these tasks.

Importance of dataset size. In Fig. 4(c), we report disagreement across test set sizes, while averaging over datasets and evaluation metrics. It is reassuring to observe that with larger test sets, the different mechanisms tend to agree more, such that it matters less which one is actually chosen. However, for MEAN vs. BT and MEDIAN vs. BT, the disagreement does not continue to decrease below 10% with more test instances. For MEAN and BT the disagreement is lower but exhibits the same behavior, never falling below a certain threshold.

Different perspectives on uncertainty. In standard evaluation setups, not only system scores are reported but also whether the differences are statistically significant (Dror et al., 2018). Therefore, we ask how often differences that are statistically significant for one test are also statistically signif-

icant for another. The details of this experiments are presented in Appendix D and show, perhaps unsurprisingly, different behavior for different tests. In particular, the paired t-test is the one that most often finds differences to be significant (for 41% of pairs); Mood's test, an unpaired test to compare medians, finds significance for only 21% of pairs; and the sign test and Wilcoxon's sign-rank test (related to BT) are in between (for 35% and 40% of the pairs, respectively).

Sources of disagreement. Based on the analysis of Sec. 3, we know that the difference between MEAN and MEDIAN is due to the presence of statistical outliers, while the difference between MEDIAN and BT is due to the presence of different test instance types (Fig. 3). With real NLP datasets, in Fig. 4, we observe some discrepancy between MEAN and MEDIAN, indicating the presence of outliers. There is even more disagreement between MEDIAN and BT, indicating the presence of different types of test instances, as illustrated in Fig. 3.

6 Related work

Several studies have made a critical assessment of the standard evaluation methodologies. For example, Freitag et al. (2020) demonstrate the advantages of carefully choosing which references to use for NLG evaluation. Mathur et al. (2020) show that outliers matter in practice. Recently, Graham et al. (2020) draws attention on test set size. Several works have emphasized the importance of careful statistical testing (Rankel et al., 2011; Owczarzak et al., 2012; Graham, 2015; Dror et al., 2018). They recommend *paired* statistical tests. Finally, Novikova et al. (2018) report that "relative rankings yield more discriminative results than absolute assessments", which further motivates aggregation mechanisms like BT.

Aggregations. Pairwise comparison mechanisms date back to Thurstone (1927). Subsequently, the Bradley-Terry (BT) model has become a standard pairwise comparison model (Bradley and Terry, 1952). In NLP, BT-inspired mechanisms have sometimes been used to aggregate human assessments. For instance, Deriu et al. (2020) ranked chatbots regarding their ability to mimic conversational behavior of humans. Item response theory (IRT) has a similar formulation as BT, but also estimates the difficulty of each test instances using a latent-variable Bayesian model (Dras, 2015).
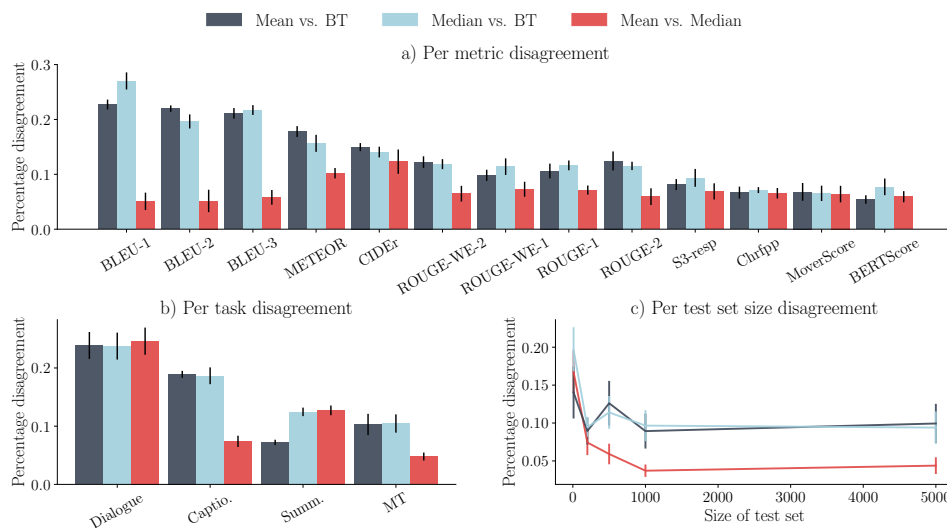
Figure 4: This figure measures the percentage of disagreement between each pair of aggregation mechanisms across different dimensions with real evaluation setups. Fig. 4(a) shows the disagreement per evaluation metric averaged over tasks and uniformly subsampled test set sizes, Fig. 4(b) shows the disagreement per task averaged over evaluation metrics and uniformly subsampled test set sizes, and Fig. 4(c) shows the disagreement across test set sizes averaged over tasks and evaluation metrics.

IRT has been applied to perform dataset filtering (Lalor et al., 2016, 2019), evaluate chatbots from human assessments (Sedoc and Ungar, 2020), and aggregate human assessments in machine translation (Dras, 2015). Elo (Elo, 1978) and TrueSkill (Herbrich et al., 2007) are famous extensions of the BT model commonly used to rate players in the context of gaming or sports events. Elo views player strengths as normally distributed random variables. TrueSkill is a Bayesian variant of Elo. Since 2015, the Workshop on Machine Translation (WMT) has been using TrueSkill to rank models based on human assessments following the methodology of Sakaguchi et al. (2014). We provide a detailed presentation and comparison of BT, Elo, and TrueSkill in Appendix G, and make both Elo and TrueSkill available as alternatives to BT in the released tool. The arguments in favor of BT made in this work transfer to its variants, including IRT, Elo, and TrueSkill, and the conclusions drawn from the experiments of Sec. 5 still hold when replacing BT by Elo or TrueSkill (Appendix G). Our work extends previous works that has considered BT variants by analyzing the potential causes for disagreement with MEAN and MEDIAN and by measuring the disagreement in real NLP evaluation setups.

7 Discussion

We briefly discuss some possible questions raised by the use of BT-like metrics, with more details provided in Appendix E, F, G, and H.

Extension to other evaluation setups. The experiments of Sec. 5 focus on reference-based NLG evaluation metrics. However, the arguments laid out throughout the paper apply beyond this setup. Any comparison of systems based on score aggregation is susceptible to suffer from outliers and complex pairing structures (e.g., Fig. 2). Future work should replicate our experimental setup for reference-free NLG (Zhao et al., 2020), classification, or regression tasks.

Type imbalance. Imagine a test set with a majority of easy instances and few hard ones. A system *A* could perform slightly worse than *B* on easy instances but much better on hard ones and will be declared worse by BT. If one views this decision as problematic then one should probably acknowledge that the test set is not representative of what should be measured. If hard instances matter more there should be a majority of them in the test set. Hoping that MEAN will be swayed to output the *intuitive* ordering of systems from a minority of test instances is a peculiar expectation to have about the evaluation setup. To diagnose such pathological cases, our tool, *Pairformance,* offers the possibility to view pairwise plots (as in Fig. 2) and histograms

2308

of score differences. More generally, better aggregation mechanisms such as BT do not solve all potential problems of evaluation methodologies. Other aspects (such as choosing evaluation metrics or meaningful, representative, and large test sets) are all independent of the choice of aggregation mechanism, but also critical to the quality of the evaluation.

Transitivity. BT is not computed independently for each system, and it can happen that adding or removing a baseline impacts the scores of other systems. We explain this phenomenon in Appendix F and show that it is rarely a problem in real data. More generally, we discuss the connection with Arrow's impossibility theorem in the context of the aggregation of social preferences (Arrow, 1950). The *Pairformance* tool gets around this difficulty by offering the possibility of analyzing each pair of systems independently.

Relaxing assumptions. BT assumes that the relative strengths of systems remain constant across test instances. This might not always be true, especially when some systems are crafted for some specific kind of instances but perform badly on others. In such cases, BT still produces meaningful and easily interpretable results but fails to capture the latent structure of system strengths. Several refinements of BT are possible; e.g., item response theory extends BT by modeling instance difficulty, and Elo and TrueSkill allow system strengths to be stochastic and vary across instances. These refinements come at the cost of introducing new parameters, and it remains unclear how to choose these parameters in practice. Future work should investigate systematic ways to choose these parameters.

Tool description. We release *Pairformance,* a tool for performing full diagnostic analyses based on an evaluation dataframe made of the evaluation scores of systems and baselines. It can perform the analysis based on MEAN, MEDIAN, BT, Elo, and TrueSkill. For each aggregation technique, it outputs a full pairwise analysis of all pairs of systems. For MEAN and MEDIAN it compares score differences for pairs of systems. For BT, Elo, and TrueSkill, it estimates the probability that one system is better than another. All analysis is accompanied by appropriate statistical testing. See Fig. 5 for an example based on the BT mechanism. Furthermore, the tool can plot the histogram of paired differences $X_A^{(l)} - X_B^{(l)}$, allowing for the direct iden-

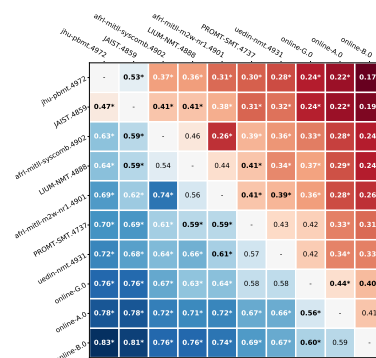|	jhu-pbmt-4972	JAIST-4859	afri-milti-syscomb-4902	LIUM-NMT-4988	afri-milti-m2w-nr1-4901	PROMT-SMT-4737	uedin-nmt-4931	online-G-0	online-A-0	online-B-0
jhu-pbmt-4973	-	0.53*	0.37*	0.36*	0.31*	0.30*	0.28*	0.24*	0.22*	0.17*
JAIST-4859	0.47*	-	0.41*	0.41*	0.38*	0.31*	0.32*	0.24*	0.22*	0.19*
afri-milti-syscomb-4902	0.63*	0.59*	-	0.46	0.26*	0.39*	0.36*	0.33*	0.28*	0.24*
LIUM-NMT-4988	0.64*	0.59*	0.54	-	0.44	0.41*	0.34*	0.37*	0.29*	0.24*
afri-milti-m2w-nr1-4901	0.69*	0.62*	0.74*	0.56	-	0.41*	0.39*	0.36*	0.28*	0.26*
PROMT-SMT-4737	0.70*	0.69*	0.61*	0.59*	0.59*	-	0.43	0.42	0.33*	0.31*
uedin-nmt-4931	0.72*	0.68*	0.64*	0.66*	0.61*	0.57	-	0.42	0.34*	0.33*
online-G-0	0.76*	0.76*	0.67*	0.63*	0.64*	0.58	0.58	-	0.44*	0.40*
online-A-0	0.78*	0.78*	0.72*	0.71*	0.72*	0.67*	0.66*	0.56*	-	0.41
online-B-0	0.83*	0.81*	0.76*	0.76*	0.74*	0.69*	0.67*	0.60*	0.59	-

Figure 5: Pairwise system comparison with BT for machine translation with ROUGE-1, as output by the *Pairformance* tool released as part of this work.

tification of pathological patterns such as those discussed above.

8   Conclusion

We performed a critical assessment of the standard NLP evaluation methodology based on averaged scores, which ignores the natural instance-level pairing of evaluation scores when comparing systems. We showed the importance of the pairing and demonstrated the advantages of paired mechanisms such as Bradley–Terry (BT) over more standard aggregation schemes such as the mean or median. The choice of aggregation mechanism matters in real evaluation setups, and we therefore recommend BT as a robust aggregation mechanism. To facilitate adoption, we release *Pairformance,* a new tool to perform full analyses of system scores using BT and two of its variants, Elo and TrueSkill.

Acknowledgments

References

Kenneth J. Arrow. 1950. A difficulty in the concept of social welfare. *Journal of Political Economy*, 58(4):328–346.

Ondřej Bojar, Yvette Graham, and Amir Kamran. 2017. Results of the WMT17 metrics shared task. In *Proceedings of the Second Conference on Machine*

Translation, pages 489–513, Copenhagen, Denmark. Association for Computational Linguistics.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Jan Deriu, Don Tuggener, Pius von Däniken, Jon Ander Campos, Alvaro Rodrigo, Thiziri Belkacem, Aitor Soroa, Eneko Agirre, and Mark Cieliebak. 2020. Spot the bot: A robust and efficient framework for the evaluation of conversational dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3971–3984, Online. Association for Computational Linguistics.

George W. Divine, H. James Norton, Anna E. Barón, and Elizabeth Juarez-Colunga. 2018. The wilcoxon–mann–whitney procedure fails as a test of medians. *The American Statistician*, 72(3):278–286.

Mark Dras. 2015. Squibs: Evaluating human pairwise preference judgments. *Computational Linguistics*, 41(2):309–317.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

Arpad E. Elo. 1978. *The rating of chessplayers, past and present*. Arco Publishing.

Ronald A. Fisher. 1935. *The Design of Experiments*. Oliver and Boyd, Edinburgh.

Markus Freitag, David Grangier, and Isaac Caswell. 2020. BLEU might be guilty but references are not innocent. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 61–71, Online. Association for Computational Linguistics.

Yvette Graham. 2015. Re-evaluating Automatic Summarization with BLEU and 192 Shades of ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 128–137. Association for Computational Linguistics.

Yvette Graham, Barry Haddow, and Philipp Koehn. 2020. Statistical power and translationese in machine translation evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 72–81, Online. Association for Computational Linguistics.

Ralf Herbrich, Tom Minka, and Thore Graepel. 2007. Trueskill™: A bayesian skill rating system. In *Advances in Neural Information Processing Systems*, volume 19, pages 569–576. MIT Press.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28, pages 1693–1701. Curran Associates, Inc.

John P. Lalor, Hao Wu, and Hong Yu. 2016. Building an evaluation scale using item response theory. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 648–657, Austin, Texas. Association for Computational Linguistics.

John P. Lalor, Hao Wu, and Hong Yu. 2019. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4249–4259, Hong Kong, China. Association for Computational Linguistics.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Chin-Yew Lin, Guihong Cao, Jianfeng Gao, and Jian-Yun Nie. 2006. An information-theoretic approach to automatic evaluation of summaries. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 463–470, New York City, USA. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

Qingsong Ma, Ondřej Bojar, and Yvette Graham. 2018. Results of the WMT18 metrics shared task: Both characters and embeddings achieve good performance. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 671–688, Belgium, Brussels. Association for Computational Linguistics.

Qingsong Ma, Johnny Wei, Ondřej Bojar, and Yvette Graham. 2019. Results of the WMT19 Metrics Shared Task: Segment-Level and Strong MT Systems Pose Big Challenges. In *Proceedings of the Fourth Conference on Machine Translation (Volume*

2: Shared Task Papers, Day 1), pages 62–90, Florence, Italy". Association for Computational Linguistics.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Shikib Mehri and Maxine Eskenazi. 2020. USR: An unsupervised and reference free evaluation metric for dialog generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 681–707, Online. Association for Computational Linguistics.

Jun-Ping Ng and Viktoria Abrecht. 2015. Better summarization evaluation with word embeddings for ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1925–1930, Lisbon, Portugal. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2018. RankME: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 72–78, New Orleans, Louisiana. Association for Computational Linguistics.

Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. 2012. An Assessment of the Accuracy of Automatic Evaluation in Summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Maxime Peyrard, Teresa Botschen, and Iryna Gurevych. 2017. Learning to score system summaries for better content selection evaluation. In *Proceedings of the Workshop on New Frontiers in Summarization*.

Maja Popovic. 2017. chrF++: Words Helping Character n-grams. In *Proceedings of the Second Conference on Machine Translation, WMT 2017, Copenhagen, Denmark, September 7-8, 2017*, pages 612–618.

Peter Rankel, John Conroy, Eric Slud, and Dianne O'Leary. 2011. Ranking human and machine summarization systems. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 467–473, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Keisuke Sakaguchi, Matt Post, and Benjamin Van Durme. 2014. Efficient elicitation of annotations for human evaluation of machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 1–11, Baltimore, Maryland, USA. Association for Computational Linguistics.

João Sedoc and Lyle Ungar. 2020. Item response theory for efficient human evaluation of chatbots. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 21–33, Online. Association for Computational Linguistics.

Louis Leon Thurstone. 1927. A law of comparative judgement. *Psychological Review*, 34:278–286.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based Image Description Evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4566–4575.

Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulleting*, 6:80–83.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Wei Zhao, Goran Glavaš, Maxime Peyrard, Yang Gao, Robert West, and Steffen Eger. 2020. On the limitations of cross-lingual encoders as exposed by reference-free machine translation evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1656–1671, Online. Association for Computational Linguistics.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578, Hong Kong, China. Association for Computational Linguistics.

	type 1	type 2	type 3	type 4	type 5
S	23	50	40	70	60
B	28	45	30	65	50

Table 2: Example of two systems S and B with their strengths $\lambda_{t_i,S}$ and $\lambda_{t_i,B}, i \in [1,5]$ associated to each type of test instances. types.

A Reproducibility

In this section, we give additional details to ensure the reproducibility of our experiments. Furthermore, the code and data to reproduce each figure and table of the main paper is available at: https://github.com/epfl-dlab/BT-eval.

A.1 Pairing examples

It is straightforward to generate examples where the marginal distribution of the evaluation scores of two systems remain unchanged even when the pairing varies.

To do so, one can define k types of test instances. For each type t_i, each system has a probability distribution of scores for this type: $\mathcal{N}(\lambda_{t_i,S}, 1)$. So for instances of type t_i, the system S has score $\lambda_{t_i,S}$ in expectation with a variance of $\sigma^2 = 1$. Similarly, another system B can have different $\lambda_{t_i,B}$ parameters. An example is given in Table 2.

Now, observe that permuting the columns of S without changing the row B leaves the marginal distribution of S and B unchanged but changes the pairing. Then, one can simply iterate over all permutations of the row S to obtain many different pairings with fixed marginal distributions.

A.2 Simulation

We discuss the synthetic data and experiments depicted in Fig. 3.

To introduce pairing issues, we create a variable number of test instance types: N_{types}. For each test type, each system has a different distribution of scores. On test type t_i, the system s_j has a normal distribution of scores: $\mathcal{N}(\lambda_{i,j}, \sigma^2)$, where we fix $\sigma^2 = 1$ throughout our experiments. For each system, the $\lambda_{i,j}$ are sampled uniformly from $[0,1]$. Depending on the values of $\lambda_{i,j}$, the score distribution of system s_j can become multimodal. When, there is only one test type, the score of each system s_j is a normal $\mathcal{N}(\lambda_j, \sigma^2)$. In that case, the pairing can be ignored and MEAN and MEDIAN are expected to work well.

For outliers, we define f as the fraction of test instances on which systems' scores are not drawn from their distribution scores. For such instances, we first draw the scores for each systems according to their distribution and then perform a random permutation, so that each system receives a score that is not sampled from its score distribution.

Then, we vary the number of systems present in the evaluation N_{sys} and the number of test instances M. Each choice of N_{types}, f, N_{sys}, and M gives a dataframe corresponding to an evaluation setup on which we can compare MEAN, MEDIAN, and BT against the *true* latent strengths of systems $\lambda_{i,j}$. The evaluation and the y-axis in Fig. 3 is then the Kendall's τ between the ordering resulting from MEAN, MEDIAN, or BT against the ordering resulting from the $\lambda_{i,j}$.

We consider the following variations for the parameters of the experiments:

- $N_{types} \in \{1, 3, 5, 10\}$,
- $f \in \{0., 0.01, 0.025\}$,
- $N_{sys} \in \{2, 3, 5, 10, 25, 50\}$,
- $M \in \{10, 30, 100, 200\}$.

In total, we have: $4 \cdot 3 \cdot 6 \cdot 4 = 288$ parameter choices. For each we sample 10 datasets resulting in $2,880$ synthetic evaluation setups.

A.3 Real data

Each of the dataset we use contains the evaluation results of a varying number of systems for a varying number of evaluation metrics:

Summarization: CNN/DM (Hermann et al., 2015): 11,432 test instances, 12 summarization systems, and 13 evaluation metrics. TAC-08: 48 test instances, 58 summarization systems, and 13 evaluation metrics. TAC-09: 44 test instances, 55 summarization systems, and 13 evaluation metrics. TAC-11: 44 test instances, 50 summarization systems, and 13 evaluation metrics. **Captioning**: MSCOCO (Lin et al., 2014): 40,504 test instances, 12 systems, and 7 evaluation metrics. **Dialogue**: Topical-Chat (Mehri and Eskenazi, 2020): 60 test instances, 5 systems, and 13 evaluation metrics. Persona-Chat (Mehri and Eskenazi, 2020): 60 test instances, 4 systems, and 13 evaluation metrics. **MT**: WMT-17 (Bojar et al., 2017): evaluated with 11 evaluation metrics, we have the following pairs: *lv-en* (2,001 instances, 9 systems), *de-en* (3,004 instances, 11 systems), *ru-en* (3,001 instances, 9 systems), *tr-en* (3,007 instances, 10 systems), and

zh-en (2,001 instances, 16 systems). WMT-18 (Ma et al., 2018): evaluated with 13 evaluation metrics we have the following pairs: *de-en* (2,998 instances, 16 systems), *et-en* (2,000 instances, 14 systems), *fi-en* (3,000 instances, 9 systems), *ru-en* (3,000 instances, 8 systems), and *zh-en* (3,981 instances, 14 systems). WMT-19 (Ma et al., 2019): evaluated with 13 evaluation metrics we have the following pairs: *de-en* (2,000 instances, 16 systems), *fi-en* (1,996 instances, 12 systems), *gu-en* (1,016 instances, 12 systems), *kk-en* (1,000 instances, 11 systems), *lt-en* (1,000 instances, 11 systems), *ru-en* (2,000 instances, 14 systems), and *zh-en* (2,000 instances, 15 systems).

The evaluation metrics considered are: BLEU-[1,2,3,4] (Papineni et al., 2002), ROUGE-[1,2,L] (Lin, 2004), ROUGE-WE-[1,2] (Ng and Abrecht, 2015), JS-[1,2] (Lin et al., 2006), S3-[pyr, resp] (Peyrard et al., 2017), CIDEr (Vedantam et al., 2015), Chrfpp (Popovic, 2017), METEOR (Lavie and Agarwal, 2007), MoverScore (Zhao et al., 2019), and BERTScore (Zhang et al., 2020). This is a total of 18 metrics.

Sub-sampling test set sizes. In experiments reported by Fig. 4 the results are averaged after re-sampling test sets of different sizes. The test set sizes used are: $[10, 50, 100, 500, 1000, 5000]$. Results broken down per dataset and per metric that does not need resampling of test set sizes is proposed in Appendix C.

A.4 Implementations

We implement BT with `scipy.org` and numpy. For the statistical tests, we use the default implementation from `scipy.org`. For Elo, we implement a wrapper around existing code: `https://github.com/ddm7018/Elo`. Similarly, for TrueSkill, we implement a wrapper around existing code: `https://pypi.org/project/trueskill/`.

B Proof of Proposition 1

Proof. We observe that the case of the MEAN and the MEDIAN are direct by definition.

$M_{A-B} > 0$ is equivalent to saying that for more than 50% of instances, $X_A^{(l)} > X_B^{(l)}$, i.e., A is better than B on more than 50% of instances. On the other hand, BT correctly gives A better than $B \iff \mathbb{P}(A > B) > \mathbb{P}(B > A) \iff \mathbb{P}(A > B) > \frac{1}{2}$, i.e., A is better than B on more than 50% of instances. So, BT is consistent $\iff A$ is better than B on more than 50% of instances $\iff M_{A-B} > 0$. □

C Disagreement breakdown

Compared to experiments in the main paper, we provide a more detailed breakdown of the disagreement in Table 3.

D Different view on uncertainty

As argued in the main paper (Sec. 3.2), the choice of aggregation mechanism bears strong similarities with the choice of statistical test. Thus, we measure in how many setups difference between systems that are statistically significant according to one test are also significant according to another.

We compare: paired t-test (usually to compare means), the Mood's median test, and the sign test (consistent with BT). We also add the Wilcoxon sign-rank test as it was often recommended by previous work (Owczarzak et al., 2012; Dror et al., 2018).

In Fig. 6, we plot the frequency with which test j yields a significant difference among the pairs of systems for which the test i has already yielded a significant difference. The diagonal depicts the overall percentage of pairs of systems for which the test finds a significant difference. Note that the matrix is not symmetric.

Interestingly, when the Mood's median test says the difference between two system is significant, 98% of the times it is also the case for the paired t-test and 89% of the times it is also the case for the Sign test. So the Mood's median is the most restrictive, finding less often significant difference than the other two. In comparison, the Sign test and the Wilcoxon's sign-rank test find significant differences between systems much more frequently. In general, the paired t-test is the one finding differences the most frequently.

E Details about the Bradley–Terry model

Given a pair of systems S_i and S_j, the Bradley–Terry model estimates the probability $p_{i,j}$ that the system S_i is better than the system S_j based on their relative strengths: $\frac{\lambda_i}{\lambda_i + \lambda_j}$.

BT estimates these parameters λ_i for each of the n systems from the observed results of evaluation. We denote as $\omega_{i,j}$ the number of instances for which S_i scores higher than S_j. Note that, in our setup, there is one comparison per test instance. In the main paper, we said that the solutions for $\hat{\lambda}$ are found in closed-form for $n = 2$. When the number of systems is greater than 2, the parameters are

	BLEU			ROUGE			ROUGE-WE			MoverScore			BERTScore		
	Mean/BT	Med/BT	Mean/Med	Mean/BT	Med/BT	Mean/Med	Mean/BT	Med/BT	Mean/Med	Mean/BT	Med/BT	Mean/Med	Mean/BT	Med/BT	Mean/Med
TAC08 Disagree.	.09	.13	.15	.07	.13	.14	.12	.06	.13	.05	.11	.12	.05	.11	.12
TAC08 ≠ SotA	.43	.73	.47	.33	.52	.47	.58	.20	.47	.10	.50	.47	.13	.17	.27
TAC08 ≠ Top3	.73	.77	.77	.61	.80	.81	.87	.65	.80	.43	.73	.70	.60	.93	.87
TAC09 Disagree.	.08	.13	.13	.08	.16	.16	.07	.15	.16	.06	.14	.13	.06	.12	.12
TAC09 ≠ SotA	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
TAC09 ≠ Top3	.70	.70	.70	.63	.87	.82	.48	.73	.75	.33	.70	.70	.43	.73	.67
TAC11 Disagree.	.07	.12	.12	.06	.13	.12	.05	.13	.12	.04	.11	.10	.04	.11	.10
TAC11 ≠ SotA	.37	.67	.50	.42	.64	.61	.33	.67	.65	.40	.63	.63	.27	.73	.63
TAC11 ≠ Top3	.73	.87	.83	.58	.88	.87	.60	.93	.92	.57	.87	.80	.43	.87	.83
CNN/DM Disagree.	.14	.17	.12	.08	.07	.02	.06	.05	.02	.07	.06	.08	.08	.08	.04
CNN/DM ≠ SotA	.53	.80	.83	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
CNN/DM ≠ Top3	.97	.97	.90	.73	.49	.24	.90	.42	.48	.00	.00	.00	.90	.90	.06
WMT17 Disagree.	.07	.08	.05	.07	.07	.04	.07	.08	.04	.03	.04	.03	.03	.04	.03
WMT17 ≠ SotA	.17	.19	.14	.28	.42	.23	.35	.40	.19	.22	.15	.24	.15	.22	.24
WMT17 ≠ Top3	.43	.57	.40	.56	.63	.29	.57	.67	.40	.26	.37	.37	.23	.27	.33
WMT18 Disagree.	.09	.09	.03	.11	.11	.04	.12	.12	.04	.06	.06	.04	.06	.06	.03
WMT18 ≠ SotA	.67	.63	.24	.55	.65	.26	.61	.67	.66	.47	.49	.18	.43	.47	.31
WMT18 ≠ Top3	.77	.74	.25	.56	.69	.39	.66	77	.40	.57	.58	.33	.57	.58	.19
WMT19 Disagree.	.07	.08	.04	.10	.11	.04	.11	.11	.05	.05	.04	.05	.04	.04	.05
WMT19 ≠ SotA	.32	.36	.25	.44	.45	.18	.46	.48	.16	.32	.25	.33	.31	.17	.35
WMT19 ≠ Top3	.54	.42	.30	.48	.54	.30	.51	.54	.33	.54	.41	.46	.39	.26	.39
TC Disagree.	.26	.22	.34	.24	.19	.24	.27	.28	.22	.28	.19	.29	.18	.24	.20
TC ≠ SotA	.53	.43	.66	.52	.46	.40	.53	.63	.45	.63	.33	.53	.30	.40	.27
TC ≠ Top3	.57	.60	.63	.57	.56	.60	.62	.55	.47	.63	.60	.60	.53	.57	.57
PC Disagree.	.28	.24	.32	.25	.23	.22	.21	.22	.22	.12	.20	.19	.13	.12	.13
PC ≠ SotA	.50	.50	.63	.42	.53	.43	.28	.33	.30	.33	.47	.50	.30	.37	.43
PC ≠ Top3	.33	.33	.43	.42	.60	.55	.37	.72	.63	.23	.30	.27	.27	.20	.07
MSCOCO Disagree.	.20	.18	.12	.18	.14	.03	-	-	-	-	-	-	-	-	-
MSCOCO ≠ SotA	1.0	1.0	.00	.03	.03	.00	-	-	-	-	-	-	-	-	-
MSCOCO ≠ Top3	1.0	1.0	.17	1.0	1.0	.47	-	-	-	-	-	-	-	-	-

Table 3: Disagreement between aggregation mechanisms per dataset and per metric.

Figure 6: In this matrix, the cell in row i and column j indicates the frequency with which the test j finds a difference significant among the pairs of systems for which the test i has found the difference significant. For example, when the Mood's median test finds a significant difference between a pair, 98% of the times, the paired t-test also finds the difference significant.

found by an iterative optimization algorithm that maximizes the following log-likelihood:

$$\mathscr{L}(\lambda) = \sum_{i=1}^{n} \sum_{j=1}^{n} \omega_{i,j} \log(\lambda_i) - \omega_{i,j} \log(\lambda_i + \lambda_j), \quad (2)$$

where $\lambda = [\lambda_1, \ldots, \lambda_n]$.

Denote W_i as the number of comparison in which system i is better: $W_i = \sum_j \omega_{i,j}$. Then, the algorithm iteratively performs the following two updates (at step t):

$$\hat{\lambda}_i = W_i \left(\sum_{i \neq j} \frac{\omega_{i,j} + \omega_{j,i}}{\lambda_i^{(t)} + \lambda_j^{(t)}} \right)^{-1}, \; \forall i, \quad (3)$$

$$\lambda_i^{(t+1)} = \frac{\hat{\lambda}_i}{\sum_k \hat{\lambda}_k}, \; \forall i. \quad (4)$$

It can be shown that starting from a random λ this algorithm improves the log-likelihood at every iteration and converges to a unique maximum.

For the practical implementation, only a threshold ϵ defining when to stop has to be decided. We choose to stop iterating when at step t, if the new vector of parameter λ remains close to the previous one: $\|\lambda^{(t+1)} - \lambda^{(t)}\|^2 < \epsilon$. Throughout our experiments, we always set $\epsilon = 1 \cdot 10^{-9}$.

F Transitivity with BT and Arrow's theorem

One possibly counter-intuitive behaviour of BT is that adding or removing a baseline can impact the scores and ordering of other systems. For example, consider two systems A and B with the following scores: $\mathscr{M}_A = [1, 2, 3]$ and $\mathscr{M}_B = [2, 3, 1]$. Then, BT identifies system B has better with a relative strengths of $\frac{2}{3}$. Now suppose another system C is added with scores $\mathscr{M}_C = [3, 2, 1]$, running BT on these 3 systems together gives the result that all systems have an equal strength, so now B is not seen as better than A.

We search for triple of systems which exhibit this pattern in our data and couldn't find any as long as we use more than 10 test instance.

Can we hope to fix this weakness? Arrow's impossibility theorem says no (Arrow, 1950). Our setup matches very well the problem of aggregating social preferences from voters. In this context, Arrow (1950) proved that no aggregation mechanism with more than 2 voters and 3 possibilities can simultaneously meet the 3 following criterion: (i) monotonicity: if every voter prefers X over Y, then the aggregation ranks X above Y, (ii) (IAA) the aggregated preference between X and Y should remain unchanged if voter preferences between other pairs change, and (iii) no dictators: the outcome is not decided by a single voter. In our framework, voters are test instance and preferences are given by the evaluation metrics. BT can fail on the second criteria, and MEAN and MEDIAN can be dictatorial (as seen in the paper). A way around this problem is to remain with pairwise comparisons of systems $n < 3$ and use BT. In that case, there is no possibility for BT to fail on IIA.

G Variants of BT: Elo and TrueSkill

BT has been extended in various ways. We discuss here two important variants that we incorporate in our analysis tool: Elo and TrueSkill.

G.1 Elo ratings

The Elo rating (Elo, 1978) is variant of the BT with an online update rule, i.e., the rating of systems (players) is updated as new test instances (new games) arrive. As BT, Elo computes the probability that systems S_i beats system S_j. Now, the t-th test instance arrives and system S_i receives the score s_i and system S_j receives the score s_j. We update the rating R based on this observed difference $\delta_{i,j}$:

$$R_k^{(t+1)} = R^{(t)} + K\left(\delta_{i,j} - \frac{Q_i}{Q_i + Q_j}\right), \quad (5)$$

where K is parameter that has to be chosen, R the rating of some system, and Q plays a role analogous to λ_k in BT. K controls how much each new instance can change the ratings. It can be shown that, implicitly, Elo corresponds to a version of BT where the strength of systems is represented by a normal distribution: $\lambda_i + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, with a variance σ^2 shared by all players (Elo, 1978). In our implementation, we provide the user with the ability to choose K and set it to 20 by default.

	Disagree.	$\neq$ SotA	$\neq$ Top-3
MEAN vs. MEDIAN	4%	18%	30%
MEAN vs. BT	9%	40%	49%
MEDIAN vs. BT	9%	41%	55%
MEAN vs. Elo	20%	55%	84%
MEDIAN vs. Elo	19%	56%	84%
MEAN vs. TrueSkill	18%	44%	76%
MEDIAN vs. TrueSkill	17%	46%	79%
BT vs. Elo	16%	38%	75%
BT vs. TrueSkill	18%	53%	72%
Elo vs. TrueSkill	18%	45%	71%

Table 4: Global disagreement (as in Table 1) between aggregation mechanisms repeated with Elo and TrueSkill.

G.2 TrueSkill

TrueSkill (Herbrich et al., 2007) is Bayesian variant of the Elo rating system. It also updates the ratings of systems online, i.e., ratings change as new test instances arrive. Now, the strength of a system S_i is represented by a normal distribution, $\mathcal{N}(\lambda_i, \sigma_i^2)$. In contrast to Elo, each player has its own variance. The update follows Bayes rule, but is intractable in general, so message passing approximation are often employed.

H Comparison of Elo, TrueSkill, and BT

We repeat the experiments of Table 1 from the main paper by replacing BT with Elo and TrueSkill with their default parameters. The results are shown in Table 4. With Elo and TrueSkill, the same conclusions from the main paper hold, i.e., paired aggregation mechanisms exhibit significant disagreement with MEAN and MEDIAN. Some discrepancies between BT, Elo, and TrueSkill remain which calls for further investigations about which one to choose.