The pitfalls of competition

Benchmarking in machine learning and its problems

Lasse Becker-Czarnetzki

czarnetzki@cl.uni-heidelberg.de

Abstract

In the last decades the development and comparison of new machine learning systems in their prevalent fields of research (e.g Computer Vision, Natural Language Processing) was majorly driven by benchmarking and competitions. Many repositories for such competitions were built (e.g OpenML [26], WEKA [13], which made it possible for fair, transparent and objective comparisons of machine learning systems to exist. This was no doubt one driving force for the huge performance increase for many tasks and problems, that machine learning systems are trying to solve. Though in today's research it is near unthinkable to publish without reporting on benchmark task or by competing in one of the many annual competitions, there are still a lot of uncertainties and mistakes present in the way competitions are organized or researchers use benchmarks to report their systems performance. In this report i thus will talk about the most important mistakes in benchmarking competitions that are being made by organizers and participants and present some tools and best practices necessary for scientific working with benchmarks.

Contents

1 Introduction

In machine learning, most prevalent in a supervised learning setting, problems are approached by taking dataset(s), that address the problem, and training and evaluating a system(s) on that data. A benchmark is nothing else than such a dataset, which is publicly available for researchers to use and compare their systems. A benchmark competition is a centrally organized competition in which the organizers control the data being used and the evaluation process and possibly give and objective ranking of machine learning systems that participated in the competition. The term benchmark is often used synonymously for a benchmark competition but can also refer to a benchmark datasset. One competition might have multiple tasks, that are all processed by each participant of the competition. In this report i will look at benchmarking and competitions in the context of machine learning.

First i will look at the reason benchmarks are theoretically superior for objective and sound academic research and under what intentions and principles they are supposed to use (see section 2).

Second i will introduce some important tools and best practices central for a sound use of benchmarks (see section (3).

Third i will expound some major problems concerning benchmarking competitions in practice (see section 4).

Fourth i will look at the use of benchmarks beyond competitions by examining an example problem in machine translation about the unclarity of BLEU score reporting (see section 5).

Finally i will discuss the consequences for the research in benchmarking (see section 6).

2 Benchmarks, what are they about?

2.1 Why use them?

The desired goal of competitions and the use of benchmark datasets and tasks is to provide an environment that allows objective and fair evaluations and comparisons of machine learning systems. Before the time of public datasets and shared tasks researchers would often have to use their private data to evaluate their systems. This of course causes many problems. The data might often not be publicly available for other researchers to compare themselves to the originaly published work. This would also hinder reproducibility and therefore the validity of published results. The results and data can be heavily biased and the overall lack in transparency hinders the academic exchange and progress. The promised solutions for these problems is the centrally organized public competition. In such a competition an organizer team releases one or multiple public benchmark datasets, that are prepared by them and thus hopefully should fulfill certain quality standards (see section 3), and creates one or multiple tasks, which the participants are challenged with solving using the given data and machine learning. The final evaluation and comparison of all participants' systems is in the responsibility of the competition organizers. This competition design focuses on creating identical and fair conditions for every participant to avoid biases, mistakes and give a public objective comparison of current machine learning systems, architectures and techniques. This leads to a well controlled and efficient exchange, good practices and methods and establishes a objective state-of-the-art and thereby accelerates the research progress and quality. On the other hand it also creates a platform for researchers to be seen and acknowledged and advance their scientific career by getting published.

2.2 The principles of benchmarks

Benchmark competitions as conceptual solution alone of course does not guarantee the above promised perfect objective comparison of systems.
To come close to such an ideal a few principles of benchmarks need to be followed. The idea of such outlining prerequisites can already be seen in the literature in 1995 [22].
The three main principles organizers and in part the participants of benchmark competitions need to concern themselves with are validity, reproducibility and comparability.

Validity is the concept of having a objectively correct experimental design. It is the guarantee that all results are achieved in a controlled fair, standardized and statistical sound way. To achieve internal validity it is often necessary to control for confounding factors to make a isolated investigation of the cause effect relationship, that a task is about, possible. To follow this principle widely used methods like e.g a training, test split (see **??**ection train$_{t}est_{s}plit)andcheckforstatisticalsignificance(seesecton3.4)areanabsolutemust.$

Reproducibilitydemandsthattheresultsofapublishedexperimentcanbereplicatedbyotherindependentscien
Thisisjustanincompletelistofquestions,thatallneedtobeansweredtohaveachanceatreproducibiltybutarein

Comparabilityistheunderlyingprinciplebenchmarkingcompetitionareaimingtoachieve.Theobjectiveand

3 Best-, necessary practices for benchmarking

To fulfill the above described principles in practices a few necessary and best practices need to to be used by every researcher that uses a benchmark to compare a machine learning system to its competition.

3.1 Holdout, Train, test split

The objective comparison of machine learning systems that inevitably use a finite set of data points to train and evaluate is a major problem. The absolute performance for e.g a classifier can only be approximated by some accuracy/performance metric on a limited number of data examples. Thus this approximation must have the property of generalizability and be free of biases. Because no finite dataset will ever be free of biases or be a perfect generalization, benchmarks comparison requires that the evaluation is done on the same data for every participating system. In practice this is done by the so called train, test split. The idea is that for a finite dataset a specific percentage of the data is held out and is not used for training at all. In fact the machine learning system is not allowed to see this data in the experimentation and development phase at all. This portion of data is only used in the final step to evaluate the system and check for its actual generalized performance. After the system is evaluated on this so called test set it is not allowed to change any aspect of the model. In reality the data is often split in not only two but three partitions. A train, development, and test set. A common split percentage is (60, 20, 20). The experimental workflow is as follows:
In the training phase the train data is used to let the model learn some representation of

the task. Because the goal is to create a system that can generalize its ”understanding” of the problem beyond the seen training data, the development data is used to evaluate the model in the experimental phase. In the context of neural systems development data is often used to tune hyperparameters. This cycle of training on train data, evaluate on dev data, change some parameters, repeat process, can theoretically be repeated infinitely since the dev data is not used for the final evaluation and thus no validity violation is created. The last step is of course evaluating the system on the hold out test data after every parameter of the model is fixed.
To control for a violation like data snooping [1], which is the process of showing the model the held out test data before its parameters are fixed, the general process in benchmarking competitions is organized as follows. The data train, test split is done by the competition organizers before the data gets released. Tipically the train data gets released separately from the test data months before the evaluation phase. The participants can split the train data in train and dev set but can’t access the test data in any way. Only after the training/experiment phase is over and the fixed models get send in to the competition organizers the test data is released or evaluation is done by organizers themselves.

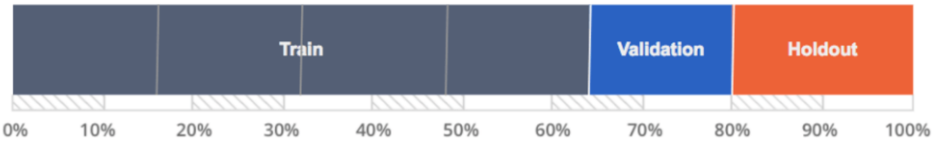

Figure 1: Train, dev, test split

3.2 Additional data handling

In this section i will look into some additional concepts that can help to control for confounders in the data or obvious biases.

Controlled use of training data
Above i already talked about the necessity of using the same test data to evaluate systems to ensure comparability. One common trend in machine learning is that the use of additional training data can often improve a system far more significantly than any learning or architecture tricks. This can have multiple reasons and factors. The amount of training data is most often very influential for generalization but also using data of other domains etc. can benefit the system. For example in recent years the method of data augmentation and use of silver data deem to be very effective. This can be a problem, if the intended research question of a benchmark competition is not to exploit these methods and data scaling, but to find the best learning methods and architectures for a given task or tasks.
To control for this a competition can forbid any use of additional data or data augmentation etc. Independent of this condition being used or not, reporting if additional data is used is essential for reproducibility. This is a prevalent problem (see section 4). In the context of Natural language processing a similar problem can be seen for data preprocessing. Often the type of preprocessing can be more important than the

machine learning model settings. It can also influence common evaluation metrics and make objective comparison near impossible (see section 5). Thus preprocessing can be pretended by the competition organizers and participants are committed to not change the data preprocessing.

Group partitioning
One common mistake which is easily explained in the case of a classification problem, is to not control for biases in the data that are created by an unbalanced class distribution. If for example one class occurs 10 times more often than every other class this creates a imbalance in the data. If one randomly splits a finite dataset into train, dev and test set this imbalance might not be represented equally in each partition. Ergo a problem of non generalizability can arise on the training or even evaluation side. To not run into this problem one can distribute the classes (groups) equally into the data partitions.

3.3 Cross Validation

A often occurring problem in practice is the lack of sufficient training data. This is a particular great problem for deep neural networks that tend to run into obstacles like overfitting, which hinders generalizability immensely. For this reason prevalent regularization techniques like dropout [25] [29] or label smoothing [19] were developed. These often require careful tuning. Therefore the goal of cross validation is to use the available training data as efficiently as possible.

K-Fold Cross Validation
The basic idea is to split the training data into k folds (see Figure 2 for a visualization). Most commonly 10-Fold cross validation is used. For one experimental iteration one uses 9 of these folds as training data and uses the 10th to evaluate the system. In other words the 10th fold is the dev data. For the next iteration the dev set fold gets switched till every fold was once used to evaluate. This way the whole training data can be used to tune hyperparameters to best generalize on the data. After all these experiments are done one has to fix the parameters before doing one more training run on the training data before finally evaluating on the hold out test set, that hasn't been seen by the system yet. To hold out this test data is crucial to not have done a form of data snooping and i want to explicitly point this out here since this can be easily misunderstood when first coming across cross validation.

Stratified Cross Validation
This form of cross validation is very similar to the latter but takes for example the class distribution in the dataset to account. This very much follows the same idea as group partitioning does and controls for imbalances in the dataset to be equally distributed in the folds (see Figure 3).

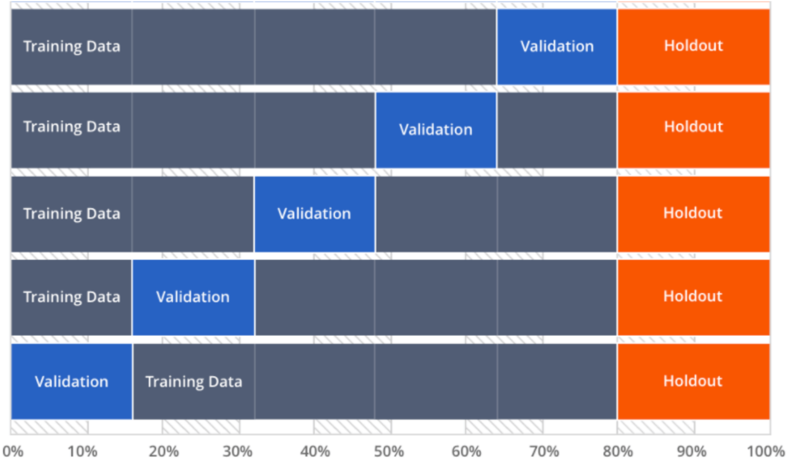

Figure 2: 5-Fold Cross Validation

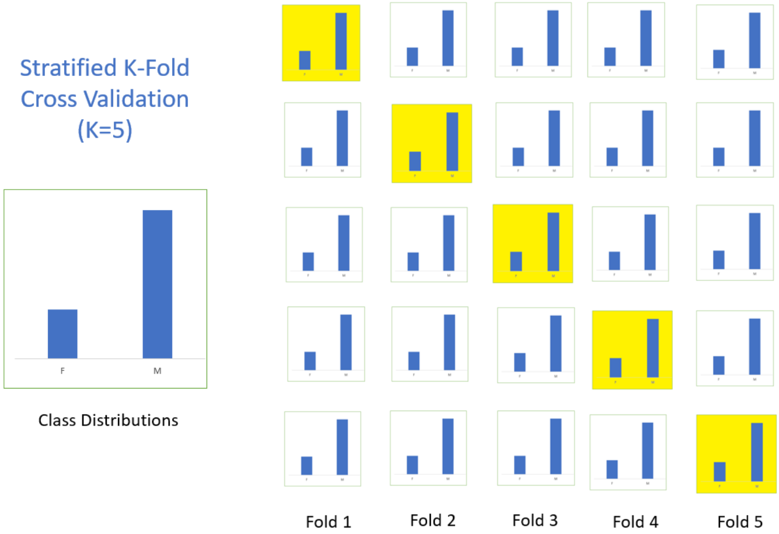

Figure 3: Stratified Cross Validation

3.4 Statistical tests

Since a finite number of data examples is always just a sample from an underlying distribution a sound comparisons of systems demands statistical tests.

Statistical significance tests are, to break it down to its core, a way of verifying if differences in system performance are actually meaningful and indicate that one model is better than another. This is important and highly underused tool in benchmarking. For a detailed description on how this is done see Hoffmann et al. [12].
It is important to know that this widely excepted method is not without major flaws and is to be taken with a grain of salt (see e.g [4] [5] [14].

Bootstrap resampling allows assigning measures of accuracy (defined in terms of bias, variance, confidence intervals, prediction error or some other such measure) to sample estimates. This can for example be used to determine robustness of rankings in regards to certain variables and makes it thus possible to quantify certainties about benchmark results. For a detailed description see HALL and MARTIN [11].

4 Problems in competitions

In this section i will look at the main flaws that benchmarking competitions have in practice, i will mainly by laying out the key points found by Maier-Hein et al. [17] who conducted a comprehensive critical analysis of biomedical image analysis challenges. The authors of [17] looked at 150 competitions including 549 tasks over 12 years and analysed interpretability and reproducibility with regards to the reported setup details of the competitions and looked at the robustness of the rankings by changing variables in the experimental setup of the competition. They find major issues in both areas. They emphasize that a great lack of information reporting is present and different changes in the experimental setups can drastically change the ranking outcome of a competition.

4.1 Lack of reported information

The authors created a list of 53 parameters they deemed important for a competition to report. 50% of the competitions didn't report 43% of these parameters. Some are very crucial for interpretability and reproducibility of competition results. 8% of the competitions that used a rank aggregation method to determine a final winner across multiple tasks did not report the specifics of the method. This is a major problem since the ranking is often not robust against different methods (see section 4.2).
85% of competitions did not report whether participants used private or public training data additionally to the data given by the competition organizers. This is highly problematic since more often the use of training data is more crucial for the success of a machine learning system than the actual system. Hence why techniques like data augmentation are common nowadays.
66% of competitions did report the details on data annotation for the gold standard. These can be very important since annotation quality varies greatly depending on annotators and annotation methods [10].
45% of tasks with multiple annotators did not describe how different annotations were handled and the final annotations were aggregated.
Those are just some examples, i refer to the paper [17] for a detailed listing of all the statistics on reported parameters. When looking at these one might think the main problem is just the interpretability but the next section shows how sensitive competitions are to some of the experiment details.

4.2 Robustness of rankings

The authors investigated how competition ranking would change if specific parameters of the experimental setup are changed. Rankings are determined by evaluation metrics on tasks and if a ranking is aggregated through multiple tasks these individual scores/rankings were aggregated in some way. They only look at relative rankings, no concrete differences in score values are given, which might be a good indicator to further investigate significance of changes. I only use the resulting findings to show how experimental parameters of competitions can influence rankings. For details i refer to the paper [**medic**]

Slight changes in metric

The quality of an evaluation can differ greatly depending on the metric being used. Maier-Hein et al. [17] show that even a slight variation in a metric can result in the last place of a ranking suddenly being first. For this they look at the Hausdorff distance(HD) [8] for evaluating image segmentation. They compare the task rankings for systems evaluated by HD and HD95, which is a slight variation that disregards extreme values and noise more. This is surely not always the case but a significant portion of tasks change their rankings and just the possibility of the last being first, depending on an often relatively arbitrarily done choice of using one of the prevalent metrics or another one, is troubling.

Rank aggregation method

If a competition includes multiple tasks and a final winner is to be determined, the different task results, possibly containing different metrics, must be aggregated. The two main methods for this are metric-based aggregation (aggregate, than rank) and case-based aggregation (rank, than aggregate).

Metric-based means one takes the mean or median, which is another parameter, of all metric results and than ranks the system based on the that.

Case-based means one assigns ranks, based on the metric, for each task and than takes mean or median of the ranks and determines the final ranking.

Both the decisions of choosing the aggregation method and using mean or median have significant impact on the ranking. They authors also used bootstrapping experiments to show, that using the mean and metric-based aggregation makes the ranking more robust. This might not necessarily be transferable to other domains of machine learning competitions but shows the need for checking for rank robustness.

Different annotators

One major flaw in reporting i talked about above is the absence of information on how many observers annotated the reference data. For tasks that did report it the authors investigated if the ranking would change, depending on which annotation was used for the reference data. If you follow the trend, it won't surprise you, that it has a significant impact on the ranking. This indicates that probably more and better annotators are needed to create a good benchmark.

Removing one test case

Above i already talked about the intention of an evaluating on an unseen test dataset to get a good spproximation on the generalizable performance of a machine learning system. For this reason the ranking of a system should not depend on the specific data being used. Unfortunately it turns out that removing just one test example can change the winner of a task, thus indicating that an objective ranking of commonly used test sets might not be valid.

4.3 Only using metric scores?

Unlike for athletes, that have to run the fastest or jump the highest, machine learning systems can't necessarily just be judged by their performance in a metric. When choosing a good system in practice often other quality measures need to be considered. In benchmarks these are sadly disregarded or not even reported.

These measures are the following:

- Model complexity
- Computational complexity
- Scalability
- Sample complexity
- Interpretability

To avoid redundancy i refer to Hoffmann et al. [12] for a detailed description of each quality measure.
The point i want to formulate here is the following. It might be necessary to investigate possibilities to report these quality measures of systems in a centralized and controlled way in benchmarking and on one hand give participants the need to properly investigate and report these measures and on the other hand give the opportunity to find a system that is maybe particularly good in one of these measures, which one might need for a specific usecase, easily, without needing to read all the signs scattered across a paper.

5 Unclarity in reporting BLEU score

In this section i will talk about how problems of incomplete or ambiguous reporting and transparency transcends different fields of research and benchmarking outside the concrete context of a competition. For this i will look at the example of unclear reporting of the BLEU score in machine translation and how this causes problems in interpretability and comparability of machine translation systems.

5.1 The BLEU score

The evaluation of a translation is without a doubt not as simple as comparing a class prediction to a gold standard. Often there are a lot of different ways a sentence can be translated into another language. May it only be some changes in the sentence structure or different words that are being used synonimously or a more literal translation versus one that is more liberal. All these variation will most likely be all marked as correct by a human evaluator. So to get a good metric based evaluation this needs to be taken into account. The Bilingual Evaluation Understudy (BLEU score) does this by computing multiple modified n-gram precisions over the proposed translation in comparison with one or more target references. For a detailed explanation see Papineni et al. [18]. Of course the BLEU score itself is not without its flaws and can be criticized (see [3], [24] but this is not the focus here.

5.2 Interpretability problems

The BLEU score is since its invention very widely used in the field of machine translation and has a great lack of competition when it comes to the metrics that are being used to compare machine translation systems on benchmark datasets. Even though in the machine learning literature one will often read something like "we used BLEU score to evaluate on..." it is not actually a fixed metric but depends on 4 parameters.

- The number of used references
- Length penalty in multi reference case
- Maximum n-gram length
- Applied smoothing to 0-count n-grams

This in itself doesn't result in a lot of gross errors in system comparison on one benchmark since almost always a maximum n-gram of four is used and most datasets only have one reference sentence. Also rarely any zero counts occur since BLEU score is corpus level. The problem lies in the fact that BLEU score is often used across multiple language pairs and datasets. The different difficulty levels for intra-language datasets of course result in highly different scores but the number of references given in a dataset also has a huge impact on the absolute score. Therefore the BLEU score of a system evaluated on a dataset with two reference sentences might be half of that of the same system evaluated on a similar dataset with four reference sentences.
This is not necessarily a widely known fact and can make wrong impressions like for example machine translation systems being way better in one language than another one, while in reality only the way the BLEU score evaluates these systems by an absolute value is different.

5.3 Comparability problems

Preprocessing the text data is an important step in machine translation to be able to give the system meaningful input. This includes modifications such as normalization (removing special characters, removing punctuation), tokenization (white space splitting of every meaningful token) and compound splitting. Concerning the BLEU score tokenization is the most important. This is because to get correct n-gram comparisons to the reference the reference also needs to be tokenized. This is why different tokenization might change the reference n-grams and therefore the BLEU score. In machine translation BLEU scores are often reported as tokenized or detokenized. This distinction refers to the way the reference tokenization is handled. Whether it is done by the user (tokenized) or internally by the metric implementation (detokenized). BLEU scores can only be properly compared if the reference tokenization is the same and because user supplied tokenization is prone to errors, this makes comparing the corresponding papers not possible. Post [20] investigated the effect of different tokenization methods on the translation tasks of the WMT'17 competition for the online-B system (see [20] for details). They found, that on average for every language a range difference of 1.0 exists. Range meaning the highest difference between the tokenization methods. The highest range found was 1.8. This differences in BLEU score are certainly higher than some reported improvements, which makes the next point very troubling.

5.4 Transparency/Reproducibilty problem

User-supplied reference processing makes the direct comparison of published scores not feasible. But if a transparent description with enough detail is provided in the paper it should be possible to reconstruct comparable BLEU scores. This is unfortunately not the case. Even in very central and famous papers in the field this level of technical detail

paper	configuration
Chiang [7]	metric$_{lc}$
Bahdanau, Cho, and Bengio [2]	(unclear)
Luong, Pham, and Manning [16]	user or metric (unclear)
Jean et al. [15]	user
Wu et al. [28]	user or user$_{lc}$ (unclear)
Vaswani et al. [27]	user or user$_{lc}$ (unclear)
Gehring et al. [9]	user, metric

Table 1: Benchmarks set by well-cited papers use different BLEU configurations. [20] ”lc” stands for lowercased

is not described in the paper or is at least unclear and not easy to determine see Table 1.

5.5 The solution

The unclear and inconsistent way the BLEU score is used in the machine translation research clearly doesn’t fulfill the principles of benchmarking explained above. The experimental setup for evaluating on benchmark datasets must be equal for every system and the corresponding paper must report they used exactly this understood method. To make this applicable in practice the reference processing must be done by the metric implementation and the user can’t change it in any way (see Figure 4).

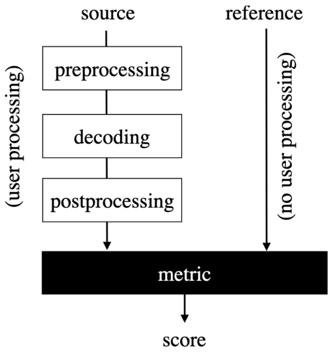

Figure 4: The proper pipeline for computing reported BLEU scores. White boxes denote user-supplied processing, and the black box, metric-supplied. The user should not touch the reference, while the metric applies its own processing to the system output and reference. [20]

The authors of [20] created a python script, which implements the BLEU score and internal tokenization, that makes this easily feasible for researchers in practice. It is called sacreBLEU and is already commonly used (see e.g [21] [6] [23]). It is still just a tool/method proposed by one paper and researchers aren’t forced to report the sacreBLEU score. It might therefore be necessary to set stricter conditions for researchers who want to publish there results on common benchmarks. Similar to the rules in actual competitions and the need to follow common practices and not perform unscientific tricks like data snooping.

6 Discussion

Benchmarking competitions play a central and important role in many fields of research in machine learning. Even though underlying principles and theoretical sound procedure is formulated these competitions/benchmarks have some key shortcomings in practice.
Reporting of competition details concerning many important parameters relevant for interpretability and reproducibility is lacking. **Robustness** of rankings are, contrary to the intention, sensitive to competition design parameters and the concept of an absolute competition winner needs to be questioned.
Narrow evaluation by just using metrics performance on datasets doesn't capture every relevant property of a machine learning system.
Unclarity in using benchmarks: The use of benchmarks for independent researchers has similar problems in lack of reporting detail and shows problems of different experimental setups, not controlled by a central competition organization, leading to not comparable results on benchmarks.
All this indicates that more elaborated guidelines for organizing competitions and using benchmarks properly are needed. Maier-Hein et al. [17] did a survey in the field of biomedical image analysis on the need for such a guideline with a clear result of great desire for it. The authors created a list of best practices to combat the problems discussed in this report. Their main contribution being the list of 53 parameters that should be reported by the competition organizers.
Their best practices can certainly be adapted to other domains and researcher fields and a similar guideline should be considered for independent researchers wanting to publish by using benchmarks. Independently of the need of such guidelines the most important lesson, as it is so often, seems to be ”think for yourself”. If you are independent researcher, combat the transparency problem and actively make complete interpretability and reproducibility of your work possible. If you are researching important state-of-the-art methods don't just use the paper that won the last competition. Look for the problems that might disturb the results and important not reported details. If necessary contact the authors to investigate validity. Don't be close minded and look for other factors than metric scores.
In conclusion the amount of problems present in benchmarking considering their importance is shocking and cries for a great expansion in research concerning these problems. This is only possible, if awareness for the situation is spread, so i request the reader to educate your fellow human beings.

7 References

[1] Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning From Data*. AMLBook, 2012. ISBN: 1600490069.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". English (US). In: 3rd International Conference on Learning Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015. Jan. 2015.

[3] Chris Callison-Burch, Miles Osborne, and Philipp Koehn. "Re-evaluating the Role of Bleu in Machine Translation Research". In: *11th Conference of the European Chapter of the Association for Computational Linguistics*. Trento, Italy: Association for Computational Linguistics, Apr. 2006. URL: https://www.aclweb.org/anthology/E06-1032.

[4] David Chavalarias et al. "Evolution of Reporting P Values in the Biomedical Literature, 1990-2015". In: *JAMA* 315.11 (Mar. 2016), pp. 1141–1148. ISSN: 0098-7484. DOI: 10.1001/jama.2016.1952. eprint: https://jamanetwork.com/journals/jama/articlepdf/2503172/joi160017.pdf. URL: https://doi.org/10.1001/jama.2016.1952.

[5] David Chavalarias et al. "Evolution of Reporting P Values in the Biomedical Literature, 1990-2015". In: *JAMA* 315.11 (Mar. 2016), pp. 1141–1148. ISSN: 0098-7484. DOI: 10.1001/jama.2016.1952. eprint: https://jamanetwork.com/journals/jama/articlepdf/2503172/joi160017.pdf. URL: https://doi.org/10.1001/jama.2016.1952.

[6] Colin Cherry et al. "Revisiting Character-Based Neural Machine Translation with Capacity and Compression". In: *CoRR* abs/1808.09943 (2018). arXiv: 1808.09943. URL: http://arxiv.org/abs/1808.09943.

[7] David Chiang. "A Hierarchical Phrase-Based Model for Statistical Machine Translation". In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. ACL '05. Ann Arbor, Michigan: Association for Computational Linguistics, 2005, pp. 263–270. DOI: 10.3115/1219840.1219873. URL: https://doi.org/10.3115/1219840.1219873.

[8] M. -. Dubuisson and A. K. Jain. "A modified Hausdorff distance for object matching". In: *Proceedings of 12th International Conference on Pattern Recognition*. Vol. 1. Oct. 1994, 566–568 vol.1. DOI: 10.1109/ICPR.1994.576361.

[9] Jonas Gehring et al. "A Convolutional Encoder Model for Neural Machine Translation". In: *CoRR* abs/1611.02344 (2016). arXiv: 1611.02344. URL: http://arxiv.org/abs/1611.02344.

[10] Katharina Grünberg et al. "Annotating Medical Image Data". In: *Cloud-Based Benchmarking of Medical Image Analysis*. Ed. by Allan Hanbury, Henning Müller, and Georg Langs. Cham: Springer International Publishing, 2017, pp. 45–67. ISBN: 978-3-319-49644-3. DOI: 10.1007/978-3-319-49644-3_4. URL: https://doi.org/10.1007/978-3-319-49644-3_4.

[11] PETER HALL and MICHAEL A. MARTIN. "On bootstrap resampling and iteration". In: *Biometrika* 75.4 (Dec. 1988), pp. 661–671. ISSN: 0006-3444. DOI: 10.1093/biomet/75.4.661. eprint: https://academic.oup.com/biomet/article-pdf/75/4/661/1170284/75-4-661.pdf. URL: https://doi.org/10.1093/biomet/75.4.661.

[12] Frank Hoffmann et al. "Benchmarking in classification and regression". In: *WIREs Data Mining and Knowledge Discovery* 9.5 (2019), e1318. DOI: 10.1002/widm.1318. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/widm.1318. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/widm.1318.

[13] G. Holmes, A. Donkin, and I. H. Witten. "WEKA: a machine learning workbench". In: *Proceedings of ANZIIS '94 - Australian New Zealnd Intelligent Information Systems Conference*. Nov. 1994, pp. 357–361. DOI: 10.1109/ANZIIS.1994.396988.

[14] John P. A. Ioannidis. "What Have We (Not) Learnt from Millions of Scientific Papers with P Values?" In: *The American Statistician* 73.sup1 (2019), pp. 20–25. DOI: 10.1080/00031305.2018.1447512. eprint: https://doi.org/10.1080/00031305.2018.1447512. URL: https://doi.org/10.1080/00031305.2018.1447512.

[15] Sébastien Jean et al. "On Using Very Large Target Vocabulary for Neural Machine Translation". In: *CoRR* abs/1412.2007 (2014). arXiv: 1412.2007. URL: http://arxiv.org/abs/1412.2007.

[16] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. "Effective Approaches to Attention-based Neural Machine Translation". In: *CoRR* abs/1508.04025 (2015). arXiv: 1508.04025. URL: http://arxiv.org/abs/1508.04025.

[17] Lena Maier-Hein et al. "Is the winner really the best? A critical analysis of common research practice in biomedical image analysis competitions". In: *CoRR* abs/1806.02051 (2018). arXiv: 1806.02051. URL: http://arxiv.org/abs/1806.02051.

[18] Kishore Papineni et al. "BLEU: A Method for Automatic Evaluation of Machine Translation". In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. ACL '02. Philadelphia, Pennsylvania: Association for Computational Linguistics, 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://doi.org/10.3115/1073083.1073135.

[19] Gabriel Pereyra et al. "Regularizing Neural Networks by Penalizing Confident Output Distributions". In: *CoRR* abs/1701.06548 (2017). arXiv: 1701.06548. URL: http://arxiv.org/abs/1701.06548.

[20] Matt Post. "A Call for Clarity in Reporting BLEU Scores". In: *CoRR* abs/1804.08771 (2018). arXiv: 1804.08771. URL: http://arxiv.org/abs/1804.08771.

[21] Matt Post and David Vilar. "Fast Lexically Constrained Decoding with Dynamic Beam Allocation for Neural Machine Translation". In: *CoRR* abs/1804.06609 (2018). arXiv: 1804.06609. URL: http://arxiv.org/abs/1804.06609.

[22] Lutz Prechelt. "Some notes on neural learning algorithm benchmarking". In: *Neurocomputing* 9.3 (1995). Control and Robotics, Part III, pp. 343–347. ISSN: 0925-2312. DOI: https://doi.org/10.1016/0925-2312(95)00084-1. URL: http://www.sciencedirect.com/science/article/pii/0925231295000841.

[23] Colin Raffel et al. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". In: *arXiv e-prints*, arXiv:1910.10683 (Oct. 2019), arXiv:1910.10683. arXiv: 1910.10683 [cs.LG].

[24] Ehud Reiter. "A Structured Review of the Validity of BLEU". In: *Computational Linguistics* 44.3 (Sept. 2018), pp. 393–401. DOI: 10.1162/coli_a_00322. URL: https://www.aclweb.org/anthology/J18-3002.

[25] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *J. Mach. Learn. Res.* 15 (2014), pp. 1929–1958.

[26] Joaquin Vanschoren et al. “OpenML: Networked Science in Machine Learning”. In: *SIGKDD Explor. Newsl.* 15.2 (June 2014), pp. 49–60. ISSN: 1931-0145. DOI: 10.1145/2641190.2641198. URL: https://doi.org/10.1145/2641190.2641198.

[27] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems 30.* Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 5998–6008. URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

[28] Yonghui Wu et al. “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation”. In: *CoRR* abs/1609.08144 (2016). arXiv: 1609.08144. URL: http://arxiv.org/abs/1609.08144.

[29] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. “Recurrent Neural Network Regularization”. In: *CoRR* abs/1409.2329 (2014). arXiv: 1409.2329. URL: http://arxiv.org/abs/1409.2329.