A Sentiment Analysis Benchmark for Automated Machine Learning Applications and a Proof of Concept in Hate Speech Detection

Marília Costa Rosendo Silva, Vitor Augusto de Oliveira, Thiago Alexandre Salgueiro Pardo

Abstract. Automated Machine Learning (AutoML) is a relevant research endeavor as it allows for speeding up and easing the development of new applied solutions using Artificial Intelligence. This paper addresses the challenge of providing standardized datasets for sentiment analysis in English and proposes an AutoML benchmark, resulting in 46 preprocessed datasets. More than this, a proof of concept is carried out for the hate speech detection task to present the potentialities of the proposed benchmark.

Resumo. O Aprendizado de Máquina Automático (AutoML) é uma área de pesquisa relevante, pois permite acelerar e facilitar o desenvolvimento de novas soluções aplicadas usando Inteligência Artificial. Este artigo aborda o desafio de fornecer conjuntos de dados padronizados para análise de sentimentos em inglês e propõe um benchmark de AutoML, resultando em 46 conjuntos de dados pré-processados. É realizada uma prova de conceito para a tarefa de detecção de discurso de ódio para apresentar as potencialidades do benchmark proposto.

1. Introduction

Natural Language Processing (NLP) aims at enabling machines to deal with human languages. The tasks of Sentiment Analysis (SA) are among the most useful and challenging ones, with interest of academic, commercial, and government areas.

In SA research, Machine Learning (ML) techniques have been the dominant approach. Developing an ML solution, however, can be complex for non-experts. For this reason, Automated Machine Learning (AutoML) has gained importance, providing resources to speed up tuning and making ML approaches more accessible [Guyon et al. 2016]. There are a few dozen available AutoML frameworks/systems, and a system that performs well on some tasks may have a lower performance on others [Škrlj et al. 2021]. Therefore, standardized comparison practices, such as benchmarks, can contribute to the traceability of the literature.

In this context, we explore AutoML for SA tasks. This work brings two core contributions: it furnishes 46 preprocessed datasets for different SA tasks; and, as Proof of Concept (PoC), some experiments with statistical evaluation to support the empirical findings comprising hate speech detection datasets and AutoML Systems.
2. Related Works
There are several initiatives on AutoML and on benchmarking some areas and tasks, but there are limited efforts focused on SA. [Blohm et al. 2021] used 13 text datasets for classification tasks, including polarity classification, with only a general evaluation. RAFT [Alex et al. 2021] is a Few-Shot Learning benchmark and uses news articles, domain-specific datasets, one Hate Speech Dataset, and another with complaints on Twitter.

Regarding comparative evaluations and statistical tests, there are several approaches in the literature and, not rarely, limited understanding of the appropriate metrics. [Demšar 2006] recommended the non-parametric Friedman test when assessing multiple classifiers in multiple datasets, and the post-hoc Nemenyi test to assess pairwise differences when the null hypothesis is rejected.

3. Dataset Collection and Preprocessing
To produce a benchmark for the SA area, it is necessary to collect and preprocess datasets, for later selecting and applying AutoML techniques, and standardizing experiment setups. Part of the procedures was based on [Pineau et al. 2020].

The data sources included UC Irvine, GitHub, Hugging Face, Kaggle, SemEval, TensorFlow, OpenML, and research articles. Our work only considered datasets without synthetically generated instances. The authors managed to split all the datasets into two non-overlapping sets (training and test sets).

The fields with text data and the one with the target feature were renamed as “text” and “label”, respectively. This standardized denomination facilitates large-scale experiments. Moreover, the preprocessing steps were customized for each dataset. In addition, the text could have more than one language. Nevertheless, this work addressed data exclusively in English that was identified with the use of fastText [Bojanowski et al. 2017]. Instances that combined English and another language were kept and Regular Expressions were used to remove Cyrillic, Chinese, or Arabic characters.

After the preprocessing steps, instances smaller than three characters or that were duplicates were excluded. Appendix A lists all the datasets and their corresponding tasks. The adopted preprocessing steps do not harm the performance of most NLP tasks.

4. AutoML Experiments
For our experiments, the task of Hate Speech Detection was chosen. It brings all the challenging characteristics of SA tasks (as subjectivity, different writing styles, and high dependence on text genre and domain) and represents a severe disease in our modern society. Handling such scientific and social problem is of utmost importance.

Thirteen preprocessed Hate Speech Detection datasets were used (see Appendix A for the citations). They were selected due to technical considerations: 10 datasets were binary, and they had different and contrasting degrees of class imbalance.

Three AutoML systems and one classifier were adopted. These systems were selected based on previous works and their underlying assumptions. The foundations of AutoGluon [Erickson et al. 2020], Auto-Sklearn [Feurer et al. 2015], and TPOT [Olson and Moore 2016] are ensembles, meta-learning, and genetic programming, respectively. The classifier was Logistic Regression (LR) combined with Random Search (RS). All the approaches had the same conditions: wall time of 15 minutes per dataset, Python Version 3.9.13, Ubuntu 22.04.2, and up to 4GB RAM.

After preprocessing all the hate speech datasets, this work embedded the text with
Figure 1. Bar Chart with Balanced Accuracy per Dataset

<table>
<thead>
<tr>
<th>Friedman</th>
<th>Kruskal-Wallis</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\chi_{n-1}$</td>
<td>$p-value$</td>
</tr>
<tr>
<td>Values</td>
<td>5.0859</td>
</tr>
</tbody>
</table>

Table 1. Statistics of the Non-Parametric Tests

Sentence BERT [Reimers and Gurevych 2019]. Figure 1 presents the bar charts corresponding to the AutoML systems and LR and their balanced accuracy per dataset.

On average, the best AutoML method is TPOT, with a mean of 71.01% of balanced accuracy. Auto-Sklearn has the largest standard deviation (14.97%). TPOT has the highest difference between the mean and median (2.24%), followed by Logistic Regression (1.82%), AutoGluon (1.60%), and Auto-Sklearn (0.61%). The maximum balanced accuracy is 86.50% (AutoGluon), whereas the smallest is 38.64% (Auto-Sklearn).

The results were evaluated with non-parametric hypothesis testing (Friedman and Kruskal-Wallis Tests). The statistics and p-values are presented in Table 1. With $\alpha < 0.05$, it is not possible to reject the null hypothesis, which means that there is no suggestion of statistically significant difference among the systems. This is a very interesting finding as it shows that different AutoML systems may prove to have similar results.

It is also interesting to evaluate the potentiality of the AutoML approaches when compared to the original results produced for the datasets. Unfortunately, it was not viable to perform comparisons for all the datasets (e.g., some of them did not have train-test splits by default). However, for three datasets, it was possible to supply fair comparisons.

Tables 2, 3 and 4 display performance metrics for three different datasets ($hs_{03}$ [de Gibert et al. 2018], $hs_{04}$ [Jigsaw 2018], and $hs_{07}$ [Zampieri et al. 2019]) using AutoML systems and manual hyperparameter tuning from related literature (the evaluation metrics are the ones of the corresponding related works). The tables compare these methods based on various metrics, using different criteria for fair comparisons. Notably, there is a discernible difference between the best results from AutoML and the reported literature, with the latter performing about 9.3% better on average. However, considering that AutoML offers a more general solution and does not require specific tuning, it proves

\footnote{sentence − transformers/all − MiniLM − L6 − v2}

sentence − transformers/all − MiniLM − L6 − v2
beneficial by relieving users of the task of building ML solutions from scratch. Despite
the performance gap, the results achieved by AutoML are deemed satisfactory, suggesting
that further investment in this approach is worthwhile.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>AutoGluon</th>
<th>Auto-Sklearn</th>
<th>LR+RS</th>
<th>TPOT</th>
<th>[de Gibert et al. 2018]</th>
</tr>
</thead>
<tbody>
<tr>
<td>hs_03</td>
<td>0.817</td>
<td>0.798</td>
<td>0.819</td>
<td>0.803</td>
<td>0.892</td>
</tr>
</tbody>
</table>

Table 2. F1-Score for Benchmarking hs_03 [de Gibert et al. 2018]

<table>
<thead>
<tr>
<th>Dataset</th>
<th>AutoGluon</th>
<th>Auto-Sklearn</th>
<th>LR+RS</th>
<th>TPOT</th>
<th>[Jigsaw 2018]</th>
</tr>
</thead>
<tbody>
<tr>
<td>hs_04</td>
<td>0.9229</td>
<td>0.9060</td>
<td>0.9278</td>
<td>0.8572</td>
<td>0.9885</td>
</tr>
</tbody>
</table>

Table 3. Accuracy for Benchmarking hs_04 [Jigsaw 2018]

<table>
<thead>
<tr>
<th>Dataset</th>
<th>AutoGluon</th>
<th>Auto-Sklearn</th>
<th>LR+RS</th>
<th>TPOT</th>
<th>[Zampieri et al. 2019]</th>
</tr>
</thead>
<tbody>
<tr>
<td>hs_07</td>
<td>0.67</td>
<td>0.67</td>
<td>0.71</td>
<td>0.70</td>
<td>0.80</td>
</tr>
</tbody>
</table>

Table 4. F1-Score Macro for Benchmarking hs_07 [Zampieri et al. 2019]

5. Final Remarks
This paper introduces a benchmark dataset for SA, performing a PoC on detecting hate
speech, showing that AutoML is a challenge that is worthy to follow. Overall, 46 pre-
processed datasets are proposed. To the best of our knowledge, this is the first work that
accomplishes this.

The proposed benchmark can be expanded with new datasets. They should have
the same rationale – train-test splits, single class per instance, the same classes in the
training and test sets, using the same codification, providing Python implementation with
all the preprocessing steps (e.g., regular expressions and sorting functions), among others
– and an available BibTex to furnish use in academia and by practitioners. These criteria
can ensure sustainable growth and an update of the proposed benchmark.

Some limitations of this work are that it comprised only English datasets and that
some datasets requiring credentials (e.g., using Twitter API to retrieve posts based on
identifiers) might lose instances due to social media policy violations. Future research
opportunities include improving algorithm initialization and evaluating other classifica-
tion strategies.

A. Datasets
The next paragraph summarizes the tasks and datasets. They are split into six SA tasks²

- **Emotion Detection** *(ed)*: [Strapparava and Mihalcea 2007], [Saravia et al. 2018],
  [Demszky et al. 2020], [Chakravarthi 2020], [Sosea et al. 2022];
- **Fake News Detection** *(fn)*: [Wang 2017], [Pérez-Rosas et al. 2018], [Torabi Asr and Taboada 2018], [Torabi Asr and Taboada 2018],
  [Thorne et al. 2018], [Abu Salem et al. 2019], [Thorne et al. 2019], [Shahi and Nandini 2020],
  [Weinzierl and Harabagiu 2022], [Weinzierl and Harabagiu 2022];
- **Hate Speech Detection** *(hs)*: [Waseem and Hovy 2016], [Davidson et al. 2017], [de Gibert et al. 2018], [Jigsaw 2018],
  [Founta et al. 2018], [Basile et al. 2019], [Zampieri et al. 2019], [Hugging Face 2019],
  [Gautam et al. 2020], [Mollas et al. 2020], [Grosz and Conde-Cespedes 2020], [Kaggle 2020c],
  [Mathew et al. 2021];
- **Polarity Classification** *(pc)*: [Pang and Lee 2005], [Go et al. 2009],
  [Maas et al. 2011], [McAuley and Leskovec 2013], [Rosenthal et al. 2014], [Zhang et al. 2015],
  [Kaggle 2020d], [Bastan et al. 2020], [Sheng and Uthus 2020];
- **Stance Detection** *(sd)*: [Kiesel et al. 2019], [Kiesel et al. 2019], [Kawintiranon and Singh 2021], [Kawintiranon and Singh 2021];
- **Utility Analysis** *(ua)*: [Grano et al. 2017], [Gräßer et al. 2018], [Kaggle 2020b], [Keung et al. 2020],
  [Kaggle 2020a].

²https://github.com/marilia-cr-silva/nlp_datasets
References


