# Evaluating Sentiment Quantification Methods in Brazilian Portuguese Corpora

Lucas Nildaimon dos Santos Silva[1][0000-0002-5606-9416], Diego Furtado Silva[2][0000-0002-5184-9413], and Helena de Medeiros Caseli[1][0000-0003-3996-8599]

[1] Graduate Program in Computer Science (PPGCC), Department of Computing, Federal University of São Carlos (UFSCar)
[2] Institute of Mathematics and Computer Sciences (ICMC), University of São Paulo (USP)
lucas.silva@estudante.ufscar.br, diegofsilva@usp.br, helenacaseli@ufscar.br

**Abstract.** This paper evaluates sentiment quantification methods applied to Brazilian Portuguese corpora. Sentiment quantification, distinct from sentiment classification, estimates the distribution of sentiment classes (positive and negative) within a dataset. We investigate several quantification techniques, including the family Classify and Count (CC) and more sophisticated methods, such as Kernel Density Estimation (KDE) and Distribution y-Similarity (DyS). Our analysis uses five datasets, each containing different distributions of sentiment classes. Our experimental results indicate that KDE and DyS methods consistently outperform others, achieving the best average ranks in terms of quantification accuracy. Statistical tests, including the Friedman and Nemenyi tests, confirm significant performance differences among the methods, with KDE and DyS showing statistically significant improvements over the baseline CC method. These findings highlight the importance of choosing robust quantification techniques for accurate sentiment quantification in corpora across different domains.

**Keywords:** Sentiment Analysis · Quantification · Distribution Shift · Brazilian Portuguese.

## 1 Introduction

Understanding the overall sentiment toward an entity is crucial for businesses, governments, public figures, and organizations. In today's data-driven world, accurately gauging public opinion helps make informed decisions, plan strategic actions, and improve public relations. With the vast amount of data generated from social media, reviews, and other digital platforms, quantifying sentiments has become essential. Effective sentiment quantification allows for the evaluation of overall sentiment, identification of trends, monitoring of changes in public perception over time, and proactive response to potential issues. Knowing the public's opinion in summary form enhances the effectiveness of these actions, as analyzing each individual opinion or comment is often impractical and inefficient.

Sentiment quantification consists of estimating the relative prevalence (or distribution) of different sentiment classes (such as positive, neutral, and negative) within a sample of unlabeled texts [8, 22]. The usual procedure for quantifying the classes of a dataset relies on adjusting the predictions made by a model induced by supervised learning. In contrast to sentiment (polarity) classification, sentiment quantification seeks to understand the overall distribution of sentiments within a dataset. Classification focuses on item-level analysis, while quantification provides an overview of sentiment trends. For instance, as illustrated in Figure 1, instead of identifying each product review as positive, negative, or neutral, sentiment quantification aims to estimate the rate of reviews that fall into each class within the dataset using quantification methods.

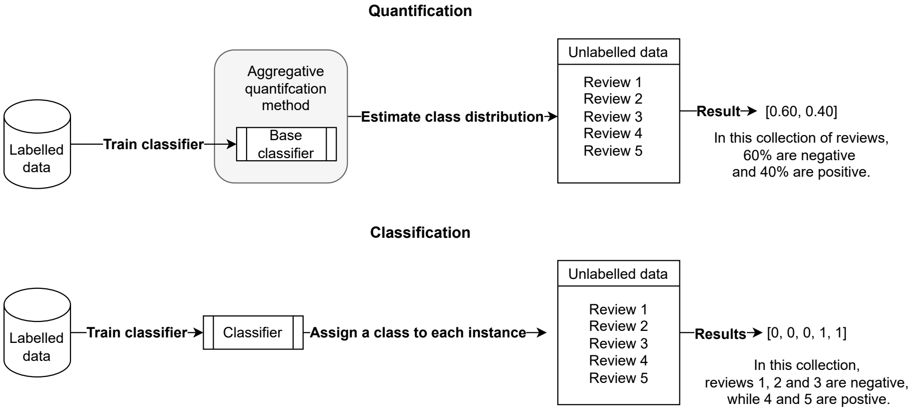


Fig. 1: Illustration of a quantification task versus a classification task with two classes.

This distinction is fundamental for applications where the overall sentiment share is more informative than the sentiment of specific instances, such as in market analysis, political sentiment monitoring, and large-scale customer feedback analysis [5].

The most straightforward approach to quantification is the Classify and Count (CC) strategy [10], which involves training a classifier to label each instance and then counting these labels to estimate class prevalence. However, most classifiers based on supervised learning assume that the distribution of the classes in the training set is identical to that in the test set, an assumption known as the Independent and Identically Distributed (IID) assumption [21]. In real-world scenarios, however, this assumption often does not hold, leading CC to inaccurate estimations of class prevalence when there are shifts in data distribution, also known as prior probability shifts. Thus, there are significant biases in class prevalence estimation when using the CC strategy [22, 19]. These limitations of the CC strategy have led to the development of quantification methods that consider prior probability shifts, such as Adjusted Classify and Count, Saerens-Latinne-Decaestecker, and methods based on distribution matching, which are investigated in this article.

Quantification methods are important because they address the limitations of traditional classification methods when dealing with changes in data distribution, such as class distribution shifts [5]. In many real-world applications, the class distribution in training data may not accurately reflect the distribution in unlabeled data. By focusing on estimating class distributions rather than classifying individual instances, quantification provides a more robust and accurate approach to data analysis in scenarios characterized by distributional shifts [22, 12].

This work presents a comparative evaluation of quantification methods in the context of sentiment analysis in Brazilian Portuguese texts. Our objective is to investigate the effectiveness of various quantification methods in estimating sentiment distribution across different datasets, including tweets and product reviews.

The main contributions of our work are:

- We provide a comparative evaluation of 11 quantification algorithms in the context of sentiment analysis in Brazilian Portuguese texts, covering diverse datasets including tweets and product reviews.
- We systematically investigate the robustness and performance of these quantification methods under varying test sizes and degrees of shift in sentiment prevalence.
- We offer insights into the strengths and weaknesses of different quantification methods, particularly highlighting their effectiveness in addressing prior probability shifts in sentiment quantification.

## 2 Related Works

The related works about quantification methods, dataset shift, and their applications cover various aspects of these topics, including proposals of algorithms, definition of experimental setups, and evaluation methodologies [9, 10, 8, 11, 16, 17, 26, 21, 22, 4, 12]. This section reviews significant contributions and methods from several key studies.

Sentiment Quantification (SQ) was initially introduced by [8]. The authors criticized the oversight in deciding whether sentiment analysis of large text collections should focus on individual or aggregate levels. They also emphasize the significance of distinguishing between sentiment classification and SQ, recognizing them as two separate applications  with distinctive characteristics. As a consequence, each of these tasks requires specific approaches. The authors claim that assuming an improvement in a classifier's accuracy at the individual level would necessarily lead to higher accuracy at the aggregate level is not valid. They illustrate this point by highlighting how the F1 Score, a widely used measure for classification evaluation, can be misleading. While a model might achieve a superior F1 Score compared to another classifier, it could achieve inferior quantification performance. To illustrate, in a binary classification task, if classifier *C1* has 20 errors (10 false positives and 10 false negatives) and classifier *C2* has 10 errors (8 false positives and 2 false negatives), *C1* would be considered worse as a classifier. However, *C1* proves to be a better quantifier than *C2* because the false positives and false negatives are balanced, compensating each other in terms of class distribution estimation. The CC method exemplifies this principle in its approach.

A comprehensive empirical evaluation of quantification methods is presented in [26]. This study evaluates 24 quantification methods across over 40 datasets, of which 3 are text data, encompassing binary and multiclass configurations. In both binary and multiclass settings, the study examined splits with varying proportions of training and test data samples. This approach simulated scenarios with both limited and abundant information available for training the models. The findings reveal that no single algorithm consistently outperforms others across all configurations. However, a group of methods, including the Median Sweep and TSMax methods based on threshold selection, the DyS framework, and the Friedman method, exhibited the best performance in the binary scenario. The best performances in the multiclass scenario were achieved by the Generalized Probabilistic Adjusted Count, the readme method, the energy distance minimization method, the EM algorithm for quantification, and the Friedman method. The study reveals that the multiclass setting poses a considerably greater challenge for established quantification methods, as evidenced by error scores that are consistently much higher than those in the binary case. Furthermore, it is shown that algorithms following the classify-and-count principle, even when optimized for quantification, tend to perform worse on average compared to other specialized methods.

The study by [12] examines the performance of existing quantification methods under various types of dataset shifts, as most previous research has primarily focused on prior probability shifts. The authors propose a taxonomy of dataset shift types and introduce experimental protocols to simulate these shifts. By testing existing quantification methods on datasets generated using these protocols, the paper aims to identify the strengths and weaknesses of these methods under different conditions. A key finding is that while many quantification methods are robust to prior probability shifts, they struggle with other types of dataset shifts. The authors introduce new evaluation protocols to simulate various dataset shifts and test several quantification methods under these conditions. Furthermore, they reveal that methods like PCC are only effective under pure covariate shifts, whereas SLD and PACC perform better when covariate shifts are accompanied by changes in class priors. However, all methods, including SLD, show instability under local covariate shifts and significant limitations in handling concept shifts. The study underscores the need for more effective quantification methods capable of addressing various types of dataset shifts.

The importance of test set size in quantification research has also been highlighted by recent studies. According to [17], the test set size is a crucial, yet often overlooked, factor in quantification research. Through empirical analysis, the authors demonstrate that the performance of quantifiers fluctuates significantly with varying test set sizes and that current methods generally perform poorly on smaller test sets. To address this, they propose a meta-learning scheme that selects the best quantifier based on the size of the test set. The authors advocate for future research to incorporate test set size considerations when evaluating new quantification proposals.

The article by [23] evaluates different quantification methods applied to sentiment data from product reviews in English and explores how quantification can enhance the accuracy of sentiment classification. The authors used six product review datasets, a pre-trained language model (Twitter-roBERTa-base), and ten quantification methods. The Artificial Prevalence Protocol (APP) was employed to generate test samples with varying class distributions to assess the error of the quantification methods and their impact on dynamic threshold adjustment for classification. The results demonstrated that eight of the nine quantification methods significantly outperformed the CC method in quantification tasks. Additionally, using quantification methods to adjust the decision threshold significantly improved classification accuracy compared to the CC method.

In contrast to the existing body of work, our study focuses specifically on the application of sentiment quantification methods to Brazilian Portuguese texts, a research problem that has received less attention. By systematically evaluating 11 quantification algorithms on five datasets of tweets and product reviews in Brazilian Portuguese, our research aims to provide a nuanced understanding of the effectiveness of these methods in this language.

## 3 Methodology

In this section, we detail the methodology employed to evaluate the effectiveness of various sentiment quantification methods on Brazilian Portuguese texts. We outline the datasets used and provide an overview of the quantification methods tested. Additionally, we explain the experimental setup, which includes multiple test sizes, $k$-fold cross-validation, a quantification-specific evaluation protocol, and the metric used to assess each method's performance.

### 3.1 Datasets

We used five datasets with texts written in Brazilian Portuguese annotated with sentiment polarity to evaluate the quantification methods. The selected datasets encompass a variety of corpora used for sentiment analysis in Brazilian Portuguese.

The “Computer-BR” dataset [18] consists of tweets related to computers and notebooks with four possible classes: irony, negative, neutral, and positive. The "Books" dataset contains book reviews manually annotated for positive and negative classes [1]. The “Sentencas” dataset [1] comprises manually labeled sentences from electronic product reviews with negative and positive classes. The

“Eleicoes2018” dataset [3] includes tweets related to the 2018 elections in Brazil, manually annotated with positive and negative polarities. The “RePro” dataset [25] consists of product reviews annotated by sentiment with four classes: positive, negative, neutral, and an ambiguous class “negative/positive.” This dataset also includes reviews topics annotation, which we disregard here.

In the experiments described in this paper, we only consider the binary tasks of sentiment quantification, excluding samples with labels different from the negative and positive classes from the corpora. The resulting class distributions vary across datasets, reflecting different proportions of positive and negative sentiments in each corpus, as shown in Table 1.

| **Dataset** | **Number of Samples** | **Class Distribution (Negative×Positive)** |
|---|---|---|
| Computer-BR | 604 | 0.67 × 0.33 |
| Books | 175 | 0.50 × 0.50 |
| Eleicoes2018 | 447 | 0.55 × 0.45 |
| Sentencas | 175 | 0.31 × 0.69 |
| RePro | 7576 | 0.46 × 0.54 |

Table 1: Summary of Datasets.

### 3.2 Quantification Methods

This study focuses on aggregative-based quantification methods, which estimate class prevalence using the output of either hard or probabilistic classifiers. The work by [11] classified quantification methods into three groups: (i) Classify, Count, and Correct, which involves classifying instances and then correcting class counts; (ii) Adapting traditional classification algorithms to function as quantifiers; and (iii) Distribution Matching, which models the training distribution and finds the best match against the test set.

The most basic approach is the CC method. This involves training a standard classifier to assign each data sample to a specific class and then counting the predicted classes to assess their distribution. However, CC often fails to estimate class prevalence accurately when there are shifts in data distribution, also known as prior probability shifts. A variant of CC is the Probabilistic Classify and Count (PCC) [2], which uses probabilistic classifiers to estimate class prevalence. Unlike CC, which relies on hard classifications, PCC considers the posterior probabilities assigned to each class by the classifier, aiming for more precise prevalence estimates. The Adjusted Classify and Count (ACC) [10] further refines the CC approach by correcting the raw counts based on the True Positive Rate (TPR) and False Positive Rate (FPR) estimated in a validation set, thus offering a more balanced estimation. The Probabilistic Adjusted Classify and Count (PACC) [2] extends ACC by integrating probabilistic classifiers.

Threshold selection techniques such as Median Sweep (MS) [9, 10], MS2 [9, 10], and MAX [9, 10] leverage the ACC method to enhance class prevalence estimation, addressing stability concerns that arise when TPR and FPR are closely aligned, particularly in scenarios where the positive class is rare. These methods employ various thresholds to adjust the classifier's decision boundary for more accurate prevalence estimation, each following strategies tailored for optimal performance. The MS method computes the median of prevalence estimates obtained by applying the ACC method across a range of classification thresholds, using cross-validation to estimate TPR and FPR, followed by the ACC correction. MS2, a stricter variant of MS, focuses specifically on thresholds where the difference between TPR and FPR exceeds 0.25, enhancing accuracy by filtering out potential outliers. Conversely, the MAX method selects a threshold that maximizes the difference between TPR and FPR, optimizing the classifier's decision boundary within the ACC framework. All previously described methods are examples of the Classify, Count, and Correct family.

An adaptation of a traditional classification algorithm to function as a quantifier, SVM(MAE) [7, 22] is a Support Vector Machine variant designed to minimize Mean Absolute Error (MAE) by explicitly focusing on the measure of error used to evaluate quantification accuracy. This method ensures the learning algorithm targets and reduces the specified error measure. SVM(MAE) is an instance of the SVMPerf framework [14], which can produce classifiers optimized for multivariate loss functions. The Saerens-Latinne-Decaestecker (SLDC) [24] algorithm also adapts a classifier by employing an expectation-maximization (EM) approach with a transductive component to refine test predictions. This iterative process updates class prevalence estimates and posterior probabilities by leveraging labeled and unlabeled data. Initially, the algorithm uses a classifier trained on labeled data to estimate posterior probabilities for the test set. In the E-step, these probabilities are used to update the prevalence estimates of each class. During the M-step, the updated prevalences refine the posterior probabilities. This process repeats until convergence.

Distribution matching techniques such as HDy [13], DyS [16], and KDEy-ML [20], operate by modeling the distribution of the training set and adjusting parameters to align this distribution with that of the test set. HDy utilizes the Hellinger distance to compare probability score distributions between the training and test sets. The Hellinger distance measures the difference between two probability distributions, providing a robust metric for comparing distributions in a high-dimensional space. HDy works by calculating the Hellinger distance for each class, allowing it to quantify the degree of distribution shift and adjust the class prevalences accordingly. DyS extends HDy by incorporating various distance functions to refine the comparison. In addition to the Hellinger distance, DyS might use metrics such as the Jensen-Shannon divergence, Bhattacharyya distance, or Earth Mover's distance, among others. By leveraging multiple distance functions, DyS can capture different aspects of the distributional changes between the training and test sets. KDEy-ML approaches the problem by modeling distributions using Gaussian Mixture Models (GMMs) and optimizing within the maximum likelihood framework. This method involves fitting GMMs to the training data to model its distribution as a mixture of Gaussian components.

KDEy-ML then aims to minimize the Kullback-Leibler (KL) divergence between the test data probability density and the training data mixture density.

### 3.3 Quantification Evaluation Protocol

Evaluating quantification methods presents a challenge due to the disparity between classification and quantification tasks. In a classification task, a dataset with $n$ data points results in $n$ test data points. However, the same dataset only provides 1 test data point in a quantification task, as observed in Figure 1.

This discrepancy arises because the quantification task aims to estimate the overall distribution in a batch of data, resulting in a single prediction for the entire batch, unlike classification, which involves $n$ predictions. Various evaluation protocols for quantification have been proposed to address this challenge, such as the Artificial Prevalence Protocol (APP) [24].

APP is a widely used evaluation protocol for assessing quantification algorithms. It involves extracting multiple samples from a test dataset with controlled prevalence values and simulating scenarios where class prevalences differ between training and test sets to evaluate the robustness of quantifiers to prior probability shifts. The steps of APP include extracting samples with predefined prevalence values, generating test samples by subsampling the positive class, applying the quantifier to estimate class prevalence, and comparing estimated prevalences with true prevalences.

A more recent protocol, the Uniform Prevalence Protocol (UPP) [6], represents a modification of APP designed to generate artificial samples with diverse class prevalence values. Unlike APP, UPP does not depend on a predefined set of class prevalence values; instead, it employs Kraemer's algorithm [27] to allow these values to vary randomly. This flexibility enables UPP to offer several advantages over APP, such as permitting users to specify the desired number of samples and facilitating the selection of any conceivable distribution vector. These capabilities are crucial for scenarios where the distribution of class prevalences is complex and not easily captured by a fixed grid of values. To conduct our experiments, we chose the UPP.

### 3.4 Quantification Evaluation Metric

To evaluate and compare our methods, we employ the Absolute Error (AE), a common measure used in quantification evaluation. This metric calculates the absolute difference between the estimated prevalence and the true prevalence, providing a quantitative assessment of accuracy for the prevalence estimates obtained from the quantification methods.

The formula for AE is given by:

$$\mathrm{AE}(p, \hat{p}) = \frac{1}{|Y|} \sum_{y \in Y} |\hat{p}(y) - p(y)| \tag{1}$$

where $p$ represents the true class prevalence, $\hat{p}$ represents the estimated class prevalence, and $Y$ is the set of classes of interest. The Mean Absolute Error

(MAE) provides a straightforward assessment of the overall accuracy of prevalence estimates, it simply returns the average absolute error between the estimated and true prevalence values for all classes of interest.

## 4 Experimental Setup

We selected an L2-regularized Logistic Regression (LR) classifier as the underlying model for our quantification methods, except for SVM(MAE), which uses SVMPerf. Logistic Regression classifiers offer well-calibrated posterior probabilities, essential for methods like PCC, PACC, DyS, and SLD [22, 12], and it is widely used in the quantification literature [22, 26, 20, 12, 4]. We begin with text preprocessing, which includes converting text to lowercase, removing accents, and eliminating punctuation and special characters. Feature extraction is performed using TF-IDF, discarding features that appear in fewer than 5 training documents. The experiment is evaluated using k-fold cross-validation (kCV) with $k = 3$.

For every unique tuple $(p(\text{positive}), p(\text{negative}))$ representing class prevalence values, where each class prevalence is uniformly drawn from the unit-simplex [27], we generate $m$ random samples, each containing $q$ documents. These samples are created to reflect the class prevalence values specified by the tuple. In these experiments, we set $m = 1000$ and evaluated multiple values of $q$ (20, 50, 100, 500) to consider the impact of test size on performance, as demonstrated in [17]. For each label $y$ (denoting positive and negative), and for each sample, the extraction is performed using sampling without replacement if there are sufficient samples in the training set; otherwise, sampling is done with replacement.

We perform hyperparameter optimization on the underlying classifiers, SVMperf for the SVM(MAE) method, and Logistic Regression (LR) for all the others. As highlighted in previous research [21], this step is important for mitigating bias in the experimentation of aggregative quantification methods. The optimization must be conducted using a quantification-oriented loss rather than a classification-oriented loss [22]. Therefore, we optimize the Mean Absolute Error (MAE) on a validation set composed of 30% of the complete training set.

To execute this optimization, we apply the Uniform Prevalence Protocol (UPP) again. We extract $m$ samples of $q$ documents each for every combination of class prevalence values from the validation set of each fold in the k-fold cross-validation (kCV). The class prevalence values are uniformly selected from the unit-simplex. In this context, $m'$ is a constant set to 200, and the value of $q'$ varies along with $q$. The optimization is conducted using the Grid Search method [15]. The parameters optimized for LR are the regularization strength $C$ (ranging from $10^{-3}$ to $10^{3}$) and class weight settings (either 'None' or 'balanced'). For SVMperf, we optimize the regularization strength $C$ over the same range.

Finally, the best model obtained through this process is re-trained on the complete training set and estimates the class prevalence values for the test set of the specified cross-validation fold. The evaluation metric used to assess the overall accuracy of the quantification methods is the MAE.

## 5 Results

In this section, we present the results of our evaluation of various sentiment quantification methods applied to Brazilian Portuguese texts. We provide an analysis of the performance of each method across different datasets under varying test sizes and degrees of sentiment prevalence shifts.

### 5.1 Test size

We begin by observing the average results for all test sizes displayed in Table 2. The results suggest that no method achieved the best performance across all datasets and test sizes. Each method showed varying degrees of effectiveness depending on the specific dataset and test size conditions.

Table 2: Average MAE for each method across all test sizes.

| **Datasets** | **cc** | **acc** | **pcc** | **pacc** | **sldc** | **kde** | **hdy** | **ms2** | **ms** | **max** | **svmmae** | **dys** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Computer-BR | 0.175 | 0.126 | 0.164 | 0.103 | 0.152 | **0.089** | 0.129 | 0.103 | 0.130 | 0.132 | 0.170 | 0.103 |
| Books | 0.205 | 0.191 | 0.201 | **0.150** | 0.190 | 0.165 | 0.193 | 0.204 | 0.180 | 0.171 | 0.209 | 0.162 |
| Eleicoes2018 | 0.091 | 0.050 | 0.091 | 0.056 | 0.067 | 0.044 | 0.051 | 0.042 | **0.040** | 0.074 | 0.090 | 0.048 |
| RePro | 0.016 | 0.014 | 0.017 | 0.013 | **0.010** | 0.011 | 0.012 | 0.022 | 0.028 | 0.014 | 0.018 | **0.010** |
| Sentencas | 0.271 | 0.168 | 0.238 | 0.144 | 0.225 | 0.185 | 0.270 | **0.130** | 0.133 | 0.134 | 0.229 | 0.216 |

We further examined the average performance of the sentiment quantification methods using the Friedman test and the Nemenyi test, as illustrated in the critical difference diagram in Figure 2. The Friedman test yielded a p-value of 0.01, leading us to reject the null hypothesis that all algorithms have the same performance at the 5% significance level. This indicates statistically significant differences in the performance of the quantification methods evaluated.

The critical difference diagram from the Nemenyi post-hoc test further elucidates these differences. It ranks the average performance scores of the methods, where lower ranks indicate better performance. From the diagram, we observe that the KDE and DyS methods achieved the best average ranks (3.2), indicating their superior overall performance across the datasets and test sizes. Notably, they were the only quantification methods that achieved statistically significant differences from the baseline CC. The PACC method followed closely, demonstrating strong performance. Methods such as MS, ACC, and MS2 exhibited moderate performance. Further down the performance scale, the SLDC, HDy, and MAX methods, showed comparatively lower performance but were still effective in certain scenarios. The PCC and SVM(MAE) methods had worse average ranks, highlighting their relative underperformance. Lastly, the CC method had the worst average rank, confirming its limited effectiveness in the contexts tested.

The variability in method effectiveness highlights the importance of choosing the right quantification method based on specific dataset characteristics and test

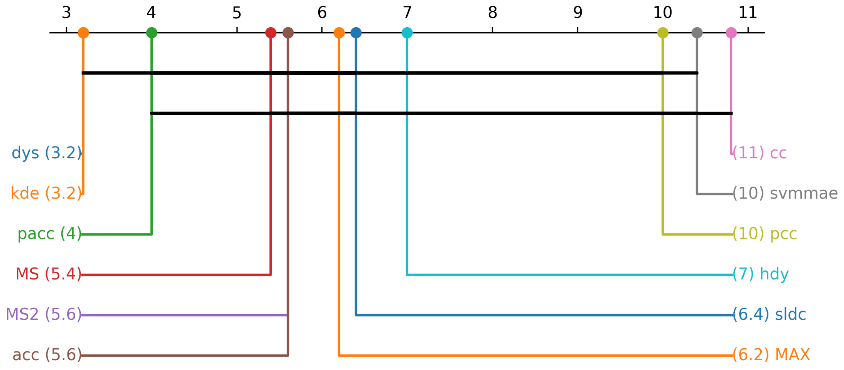


Fig. 2: Critical difference diagram of average score ranks for all test sizes.

conditions. We observed different outcomes when analyzing the critical difference diagram for each test set size. There was no statistical difference between the methods' performances for the smallest (20 samples) and the biggest (500 samples) test sizes. We could expect such behavior for the smallest test size, as evidenced by previous research [17], indicating that methods generally perform poorer on small sizes. As for the biggest size, an important factor is the original datasets' sizes and the sampling strategy used to form the 500 documents with the UPP. In this case, there was a need to apply sampling with replacement for 3 out of the 5 datasets evaluated (Books, Eleicoes2018, and Sentencas), which implies oversampling the training and test datasets. Specifically, sampling with replacement may result in certain examples being selected multiple times, which could lead to some biases, such as reducing the model's ability to generalize to truly unseen data.

We observed significant differences for test sizes of 50 and 100 samples, as illustrated in Figure 3. Overall, the diagrams highlight that KDE achieved the best performance for both test sizes and was the only method with statistically significant differences from the baseline CC. PACC and DyS consistently performed well across both test sizes, while SVM(MAE) consistently ranked lower, surpassing only the baseline.

### 5.2 Distributional shift

Continuing our analysis, we examined the performance of our method concerning the intensity of distribution shifts between training and test sets. We categorized these shifts using the Manhattan distance to measure the dissimilarity between the training and test distributions. We considered a shift major if the distance was 0.8 or higher, minor if it was less than 0.4, and medium otherwise. Figure 4 displays the methods' average MAE across all datasets for each shift intensity. As expected, the higher the shift intensity, the worse the error becomes.

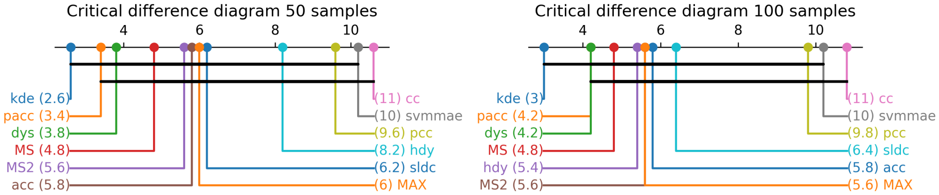


Fig. 3: Critical difference diagrams of average score ranks for test sizes 50 and 100.

Figure 5 displays the AE distribution for each dataset and shift intensity to provide a more detailed understanding of the methods' stability across shift intensities and dataset characteristics. Some methods, such as the threshold selection methods MS and MS2, as well as PACC, DyS, and KDE, exhibit relatively stable performance across shift intensities. In contrast, methods such as the baseline CC, PCC, SLDC, and SVM(MAE) displayed more unstable behavior, yielding considerably worse results as the distribution shift intensity increased. The patterns are consistent across all datasets except for RePro, where the methods demonstrated superior stability and overall better performance. This difference may be attributed to data availability, as RePro is the largest dataset with 7,576 samples. Consequently, it does not require the oversampling previously discussed to generate all sample sizes.

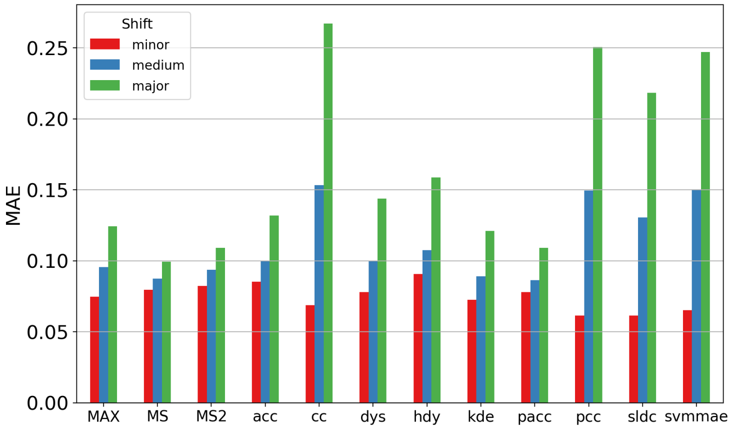
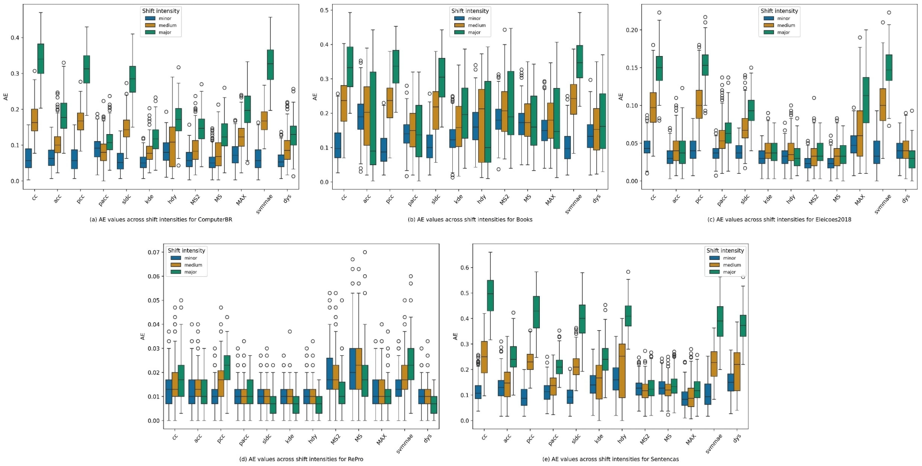


Fig. 4: Average MAE for each method across datasets and shift intensities.



(a) AE values across shift intensities for ComputerBR

(b) AE values across shift intensities for Books

(c) AE values across shift intensities for Eleicoes2018

(d) AE values across shift intensities for RePro

(e) AE values across shift intensities for Sentencas

Fig. 5: AE values across shift intensities for all datasets.

## 6 Conclusion

This study has provided a comprehensive evaluation of various sentiment quantification methods applied to Brazilian Portuguese texts under different conditions, particularly focusing on varying test sizes and the intensity of distribution shifts between training and test sets. Several findings emerged from our analysis.

Firstly, KDE consistently exhibited superior performance across different test sizes, showing statistically significant improvements over the baseline CC. Alongside KDE, PACC and DyS also performed well, demonstrating stability and effectiveness across various conditions. In contrast, SVM(MAE) consistently achieved poor results, performing only marginally better than the baseline.

The impact of distribution shift intensity on the performance of quantification methods was significant. As expected, higher shift intensities led to increased errors for all methods. However, certain methods, such as the threshold selection methods (MAX, MS, MS2), PACC, and KDE, displayed relatively stable performance, indicating their robustness to distribution shifts. Conversely, methods such as the baseline CC, PCC, SLDC, and SVM(MAE) exhibited greater instability and poorer performance as the shift intensity increased.

The Friedman test and Nemenyi post-hoc test further confirmed the significant differences in performance among the methods. KDE and DyS were the only methods that consistently achieved statistically significant improvements over the baseline across all test sizes.

Looking forward, future research could investigate the integration of deep learning techniques, such as transformer models, into quantification methods to potentially yield significant performance improvements. Understanding the trade-offs between quantification accuracy and computational efficiency will be crucial for deploying these methods in resource-constrained environments. More-over, conducting more comprehensive evaluations across a wider range of languages and domains will help generalize findings and validate the methods' applicability. Exploring semi-supervised and unsupervised methods could reduce the dependency on annotated data, making sentiment quantification more accessible and scalable across various domains and sample sizes. Additionally, examining the impact of different types of data augmentation techniques on the performance of quantification methods could offer insights into improving generalization and robustness.

In summary, our findings highlight the importance of selecting appropriate quantification methods based on the specific characteristics of the dataset and the anticipated distribution shift intensities. Methods like KDE and DyS offer robust performance across various conditions, making them particularly suitable for practical applications where different intensities of distribution shifts are expected.

**Acknowledgments.** This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

## References

1. Belisário, L.B., Ferreira, L.G., Pardo, T.A.S.: Evaluating methods of different paradigms for subjectivity classification in portuguese. In: Quaresma, P., Vieira, R., Aluísio, S., Moniz, H., Batista, F., Gonçalves, T. (eds.) Computational Processing of the Portuguese Language. pp. 261–269. Springer International Publishing, Cham (2020)
2. Bella, A., Ferri, C., Hernández-Orallo, J., Ramirez-Quintana, M.J.: Quantification via probability estimators. In: 2010 IEEE International Conference on Data Mining. pp. 737–742. IEEE (2010)
3. Cristiani, A., Lieira, D., Camargo, H.: A sentiment analysis of brazilian elections tweets. In: Anais do VIII Symposium on Knowledge Discovery, Mining and Learning. pp. 153–160. SBC (2020)
4. Donyavi, Z., Serapio, A., Batista, G.: Mc-sq: A highly accurate ensemble for multiclass quantification. In: Proceedings of the 2023 SIAM International Conference on Data Mining (SDM). pp. 622–630. SIAM (2023)
5. Esuli, A., Fabris, A., Moreo, A., Sebastiani, F.: Learning to Quantify. Springer Nature (2023)
6. Esuli, A., Moreo, A., Sebastiani, F.: Lequa@ clef2022: Learning to quantify. In: European Conference on Information Retrieval. pp. 374–381. Springer (2022)
7. Esuli, A., Sebastiani, F.: Optimizing text quantifiers for multivariate loss functions. ACM Transactions on Knowledge Discovery from Data (TKDD) **9**(4), 1–27 (2015)
8. Esuli, A., Sebastiani, F., Abbasi, A.: Sentiment quantification. IEEE Intell. Syst. **25**(4), 72–75 (2010)
9. Forman, G.: Quantifying trends accurately despite classifier error and class imbalance. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 157–166 (2006)
10. Forman, G.: Quantifying counts and costs via classification. Data Mining and Knowledge Discovery **17**, 164–206 (2008)

11. González, P., Castaño, A., Chawla, N.V., Coz, J.J.D.: A review on quantification learning. ACM Computing Surveys (CSUR) **50**(5), 1–40 (2017)
12. González, P., Moreo, A., Sebastiani, F.: Binary quantification and dataset shift: an experimental investigation. Data Mining and Knowledge Discovery pp. 1–43 (2024)
13. González-Castro, V., Alaiz-Rodríguez, R., Alegre, E.: Class distribution estimation based on the hellinger distance. Information Sciences **218**, 146–164 (2013)
14. Joachims, T.: A support vector method for multivariate performance measures. In: Proceedings of the 22nd international conference on Machine learning. pp. 377–384 (2005)
15. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998). https://doi.org/10.1109/5.726791
16. Maletzke, A., dos Reis, D., Cherman, E., Batista, G.: Dys: A framework for mixture models in quantification. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 4552–4560 (2019)
17. Maletzke, A.G., Hassan, W., dos Reis, D.M., Batista, G.E.: The importance of the test set size in quantification assessment. In: IJCAI. pp. 2640–2646 (2020)
18. Moraes, S.M.W., Santos, A.L.L., Redecker, M., Machado, R.M., Meneguzzi, F.R.: Comparing approaches to subjectivity classification: A study on portuguese tweets. In: Silva, J., Ribeiro, R., Quaresma, P., Adami, A., Branco, A. (eds.) Computational Processing of the Portuguese Language. pp. 86–94. Springer International Publishing, Cham (2016)
19. Moreo, A., Francisco, M., Sebastiani, F.: Multi-label quantification. ACM Transactions on Knowledge Discovery from Data **18**(1), 1–36 (2023)
20. Moreo, A., González, P., del Coz, J.J.: Kernel density estimation for multiclass quantification. arXiv preprint arXiv:2401.00490 (2023)
21. Moreo, A., Sebastiani, F.: Re-assessing the "classify and count" quantification method. In: Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28–April 1, 2021, Proceedings, Part II 43. pp. 75–91. Springer (2021)
22. Moreo, A., Sebastiani, F.: Tweet sentiment quantification: An experimental re-evaluation. PLoS One **17**(9), e0263449 (2022)
23. Ojeda, D.Z., Zalewski, W., Maletzke, A.G.: Utilizando a quantificação na análise de sentimentos em reviews de produtos. In: Escola Regional de Banco de Dados (ERBD). pp. 71–80. SBC (2024)
24. Saerens, M., Latinne, P., Decaestecker, C.: Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. Neural computation **14**(1), 21–41 (2002)
25. dos Santos Silva, L.N., Real, L., Zandavalle, A.C.B., Rodrigues, C.F.G., da Silva Gama, T., Souza, F.G., Zaidan, P.D.S.: Repro: a benchmark for opinion mining for brazilian portuguese. In: Proceedings of the 16th International Conference on Computational Processing of Portuguese. pp. 432–440 (2024)
26. Schumacher, T., Strohmaier, M., Lemmerich, F.: A comparative evaluation of quantification methods. arXiv preprint arXiv:2103.03223 (2021)
27. Smith, N.A., Tromble, R.W.: Sampling uniformly from the unit simplex. Johns Hopkins University, Tech. Rep **29** (2004)