# Transfer learning for Twitter sentiment analysis: Choosing an effective source dataset

E. Guimarães[1,2], J. Carvalho[3], A. Paes[1], A. Plastino[1]

[1] Universidade Federal Fluminense, Brazil
[2] Marinha do Brasil
eliseuguimaraes@id.uff.br {alinepaes,plastino}@ic.uff.br
[3] Instituto Federal Fluminense, Brazil
joncarv@iff.edu.br

**Abstract.** Sentiment analysis on social media data can be a challenging task, among other reasons, because labeled data for training is not always available. Transfer learning approaches address this problem by leveraging a labeled source domain to obtain a model for a target domain that is different but related to the source domain. However, the question that arises is how to choose proper source data for training the target classifier, which can be made considering the similarity between source and target data using distance metrics. This article investigates the relation between these distance metrics and the classifiers' performance. For this purpose, we propose to evaluate four metrics combined with distinct dataset representations. Computational experiments, conducted in the Twitter sentiment analysis scenario, showed that the cosine similarity metric combined with bag-of-words normalized with term frequency-inverse document frequency presented the best results in terms of predictive power, outperforming even the classifiers trained with the target dataset in many cases.

CCS Concepts: • **Computing methodologies** → **Transfer learning**.

Keywords: dataset representation, machine learning, metrics, sentiment analysis, supervised learning, transfer learning

## 1. INTRODUCTION

Sentiment analysis is a suitcase research problem [Cambria et al. 2017] that involves many Natural Language Processing (NLP) tasks, including the polarity classification of opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities and their attributes expressed in written text [Liu 2012]. With the explosion of social media networks, especially Twitter, people are free to express themselves on any topic using a limited number of characters in short messages called tweets. In this scenario, applying sentiment analysis is particularly challenging considering the characteristics of these short informal messages, such as the incorrect use of grammar, the presence of misspelled words, and lack of context [Martínez-Cámara et al. 2014]. Regarding the polarity detection problem, which aims at identifying whether a text conveys a positive or a negative opinion, two main approaches have been adopted in the literature: lexicon-based methods and machine learning strategies.

Lexicon-based methods rely on the prior polarity of words from existing dictionaries, or lexicons. On the other hand, machine learning strategies, which are the focus of this study, extract characteristics from labeled data in a given domain, called features, and train a model to predict the polarity of new data. However, enough labeled data is not always available, either because the target domain is rare or because manually labeling existent data requires much human effort. In that case, transfer learning approaches emerge as a feasible solution by using labeled data from a different but related source domain to train a classifier to the domain of interest, *i.e.*, the target domain [Pan and Yang 2010].

Nevertheless, choosing between all labeled datasets available from different source-related domains remains a key challenge.

In the context of the challenging issue of choosing an appropriate source dataset, this article aims at determining which metric from a set of distinct distance metrics can be used to identify the most appropriate labeled dataset from a source domain to train a classifier via transfer learning. For this purpose, we evaluate four different distance metrics to select a source dataset, combined with distinct approaches for dataset representation.

The conducted computational experiments, conducted in the Twitter sentiment analysis scenario, showed that the cosine similarity metric combined with bag-of-words normalized with term frequency-inverse document frequency presented the best results, in terms of predictive power, outperforming even the classifiers trained with the target dataset in many cases.

The remainder of this article is organized as follows. Section 2 brings some important concepts used in the article, Section 3 shows examples of similar studies in the literature. Section 4 presents the workflow of the experiments carried out in this study, Section 5 displays and evaluates the results obtained with the experiments, and 6 discusses the conclusions and indicates new research directions.

## 2. BACKGROUND

In this section, we present some definitions for helping in the comprehension of this article.

**Transfer learning**: Transfer learning allows the domains, tasks, and distributions used in training and testing to be different [Pan and Yang 2010]. Basically, it uses the source domain and a learning task in this domain to improve the learning for a task in the target domain, using the knowledge obtained in the source domain. It is grounded on the idea that appropriating from prior knowledge and learning can be useful and save resources, avoiding starting from the scratch for every new problem when labeled data is rare or not available. Recently, using transfer learning to solve natural language tasks in the presence of limited data has become a very attractive field of research [Ruder 2019; Devlin et al. 2019].

**Word embeddings**: Word embeddings [Mikolov et al. 2013] is a technique to represent words in low-dimensional real-valued vectors. Such vectors are learned from large corpora of textual data using neural network techniques aimed at capturing the word's meaning. In that case, words that are frequently used in the same context are represented in the same space.

**Cosine similarity**: Given two vectors $u = (u_1, u_2, ..., u_n)$ and $v = (v_1, v_2, ..., v_n)$ the cosine similarity (CS) between them is defined as follows:

$$CS = \frac{u \cdot v}{\|u\|.\|v\|} \tag{1}$$

where $u \cdot v$ represents the inner product between $u$ and $v$, and $\|u\|$ and $\|v\|$ represents their norms.

**Euclidean distance**: The Euclidean Distance (ED) between two vectors $u = (u_1, u_2, ..., u_n)$ and $v = (v_1, v_2, ..., v_n)$ is defined as follows:

$$ED = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2} \tag{2}$$

**Jaccard distance**: The Jaccard Distance (JD) between two sets $A$ and $B$ is defined as the complement of the ratio between their intersection size and their union size. Then:

$$JD = 1 - \frac{|A \bigcap B|}{|A \bigcup B|} \tag{3}$$

**Relaxed word moving distance** [Kusner et al. 2015]: Consider two datasets $D_i$ and $D_j$ whose word embeddings representations have $n$ and $m$ elements, respectively. The Relaxed Word Moving Distance (RWMD) can be defined as follows:

$$\text{RWMD} = max\left(\sum_{a=1}^{n} f_{ia} \times ED^*_{ia}, \sum_{b=1}^{m} f_{jb} \times ED^*_{jb}\right) \tag{4}$$

where $f_{ia}$ represents the word embedding relative frequency for the *a-th* element of $D_i$ and $ED^*_{ia}$ is the Euclidean distance between the *a-th* element and its closest word embedding in $D_j$. The terms $f_{jb}$ and $ED^*_{jb}$ are analogous. Thereby, RWMD computes the greatest cost of moving from one dataset to another, weighted by the relative frequencies of the word embeddings, considering both of them as possible origins.

## 3. RELATED WORK

Distinct studies have been presented in the literature aiming at determining an appropriate distance metric to select data in a given domain to train a classifier to a different target domain via transfer learning [Van Asch and Daelemans 2010; Plank and van Noord 2011; Remus 2012; Ruder and Plank 2017; Santos et al. 2019].

[Van Asch and Daelemans 2010] investigated the relationship between the difference of source and target datasets and the accuracy of Part-of-Speech (PoS) tagger. For the difference calculation, the correlations between six distance metrics and the accuracy of the POS tagger were used, and they showed that Rényi divergence had the best performance in predicting the accuracy of the tagger. In [Plank and van Noord 2011], they studied six metrics and two types of feature representations and their performance in helping select data for transfer learning in parsing tasks in English and Dutch. They found that the variational metric using a topic model representation was the best technique.

Differently, when target data is labeled, [Remus 2012] proposed an approach to select instances from the source dataset based on two metrics: domain similarity and domain complexity. These selected instances and the target dataset were used to compose a new source dataset. Domain similarity was considered based on the idea that selecting the most similar instances to the target dataset could aggregate more information to the trained model. In its turn, the difference between the domain complexities of source and target datasets were used to calculate the reduction to be applied in the original source dataset. The idea behind this was that the more different their complexities are, the less the source data would be useful to compose the new source dataset.

Recently, [Ruder and Plank 2017] proposed an approach to learn data selection measures using Bayesian Optimization for three tasks: sentiment analysis, POS tagging, and parsing. For that purpose, they used six distance metrics as features to learn the new measure, considering three types of dataset representations. Furthermore, they took into consideration that diversity could improve the quality of the training model. Thus, for each training instance, they calculated its diversity, believing that some of them are well suited for knowledge acquisition. The results achieved by them outperformed the existing distance metrics.

[Santos et al. 2019] evaluated three distance metrics on sentiment analysis in the domain of the 2018 Brazilian Presidential Elections using social media data, like tweets, in Portuguese. These metrics were used for datasets selection with the purpose to merge them, and they showed that choosing similar datasets helps in achieving better results. Additionally, they showed that selecting dissimilar datasets worsens the results of the classifiers.

This article differs from previous studies because it investigates, in the scenario of sentiment analysis of tweets, the relationship between distance metrics and the performance of the classifiers trained with the datasets selected by these metrics when applied to the target datasets. Also, in order to conduct our experiments, we have used a large set of 22 Twitter datasets in English.

| Dataset | Abbreviation | Positive | % positive | Negative | % negative | Total tweets |
|---|---|---|---|---|---|---|
| irony | iro | 22 | 34% | 43 | 66% | 65 |
| sarcasm | sar | 33 | 46% | 38 | 54% | 71 |
| ntua | ntu | 159 | 57% | 119 | 43% | 278 |
| SemEval15-Task11 | S15 | 47 | 15% | 274 | 85% | 321 |
| sentiment140 | stm | 182 | 51% | 177 | 49% | 359 |
| person | per | 312 | 71% | 127 | 29% | 439 |
| hobbit | hob | 354 | 68% | 168 | 32% | 522 |
| iphone | iph | 371 | 70% | 161 | 30% | 532 |
| movie | mov | 460 | 82% | 101 | 18% | 561 |
| sanders | san | 570 | 47% | 654 | 53% | 1224 |
| Narr-KDML-2012 | Nar | 739 | 60% | 488 | 40% | 1227 |
| archeage | arc | 724 | 42% | 994 | 58% | 1718 |
| SemEval18 | S18 | 865 | 47% | 994 | 53% | 1859 |
| debate08 | deb | 710 | 37% | 1196 | 63% | 1906 |
| HCR | HCR | 539 | 28% | 1369 | 72% | 1908 |
| STS-gold | STS | 632 | 31% | 1402 | 69% | 2034 |
| SentiStrength | SSt | 1340 | 59% | 949 | 41% | 2289 |
| Target-dependent | Tar | 1734 | 50% | 1733 | 50% | 3467 |
| VADER | VAD | 2897 | 69% | 1299 | 31% | 4196 |
| SemEval13 | S13 | 3183 | 73% | 1195 | 27% | 4378 |
| SemEval17-test | S17 | 2375 | 37% | 3972 | 63% | 6347 |
| SemEval16 | S16 | 8893 | 73% | 3323 | 27% | 12216 |

Table I. Datasets characteristics.

## 4. METHODOLOGY

To conduct the investigation proposed in this article, we used a set of 22 datasets[1] of tweets [Carvalho and Plastino 2020]. Table I presents some characteristics of these datasets, namely their abbreviation, number and fraction of positive and negative tweets, and total number of tweets.

We adopted the following preprocessing steps. First, for each tweet in a given dataset, we replaced URLs and user mentions by unique tokens. Then, all characters were lowercased, and the resulting tweet was tokenized. Finally, we used a pretrained embedding model [Bravo-Marquez et al. 2016], trained over ten million tweets from the Edinburgh Twitter corpus [Petrovic et al. 2010] using the Skip-gram method, to generate a representation for each tweet by averaging the embedding values of its tokens. Henceforth this representation is named as tweet embeddings. We adopted this pretrained model regarding its good performance when compared to other models [Carvalho and Plastino 2020].

To determine the similarity between datasets, we measured the distance between them using the metrics presented in Section 2, i.e., Euclidean distance, cosine similarity, Jaccard distance, and Relaxed Word Moving Distance. For the Euclidean distance, we used two types of representation: dataset embeddings as the average of all word embeddings of the dataset (ED1) and dataset embeddings as the average of all tweet embeddings of the dataset (ED2). The cosine similarity was computed using three forms of representation: bag-of-words (BoW) with term frequency-inverse document frequency (TF-IDF) (CS1), dataset embeddings as the average of all word embeddings of the dataset (CS2), and dataset embeddings as the average of all tweet embeddings of the dataset (CS3). For the Jaccard distance (JD), one more preprocessing step was needed: the lemmatization of the tokens. Then, the lemma sets were considered for the calculation. According to RWMD definition, all word embeddings of the datasets were taken into account for its calculation.

We adopted Scikit Learn's [Pedregosa et al. 2011] implementation of Logistic Regression to train the classifiers. This algorithm was chosen by its good performance in sentiment analysis in Twitter scenario [Carvalho and Plastino 2020]. Specifically, we used each dataset to generate a classification model which was then applied to classify the instances of the other 21 datasets.

[1]Datasets are available at this GitHub repository: `https://github.com/joncarv/air-datasets`

In the experimental evaluation, for each target dataset, we used classification accuracy and weighted average F-measure ($F_{\text{AVG}}$) to compare the results achieved by using the classifier trained with the most similar dataset pointed by the metrics and the results achieved by performing a 10-fold cross-validation when the target dataset is used to train the classifier itself.

Additionally, we compared the classification accuracy and $F_{\text{AVG}}$ results achieved when applying the classifiers trained with all datasets, one by one, for each target dataset. When source and target datasets were the same dataset, a 10-fold cross-validation was performed. This comparison intended to verify if some dataset can be selected as source dataset independently of its distance to target dataset with a low predictive loss.

## 5. EXPERIMENTAL RESULTS

To evaluate the effectiveness of the transfer learning approach, first, we performed a 10-fold cross-validation to induce the logistic regression model for each dataset. Table II presents the results of this evaluation in terms of accuracy and weighted F-measure ($F_{AVG}$) (second and third columns, respectively). Then, for each target dataset (presented in the rows), we conducted an experiment to identify the most similar dataset to it by using one distance metric at a time, to train a classifier and evaluate its predictive performance on the target dataset. Due to space constraints, we present only the results related to the CS1 metric (fourth and fifth columns), which is the one that has achieved the best overall results, as we shall see later. The sixth and seventh columns present the gain (in %) when the most similar dataset is used in the classification instead of the target dataset itself, in terms of accuracy and $F_{\text{AVG}}$, respectively. The results that increased the performance are presented in boldface type. Finally, the Average and St.dev. rows show the total average gain and its standard deviation, respectively.

| Dataset | $Accuracy_{10-FCV}$ | $F_{AVG-10-FCV}$ | $Accuracy_{CS1}$ | $F_{AVG-CS1}$ | Accuracy ratio | $F_{AVG}$ ratio |
|---|---|---|---|---|---|---|
| irony | 0.66 | 0.53 | 0.68 | 0.68 | **102.27%** | **129.60%** |
| sarcasm | 0.56 | 0.43 | 0.58 | 0.53 | **102.34%** | **123.55%** |
| ntua | 0.81 | 0.80 | 0.86 | 0.86 | **106.24%** | **107.85%** |
| SemEval15-Task11 | 0.85 | 0.79 | 0.70 | 0.74 | 82.47% | 94.06% |
| sentiment140 | 0.81 | 0.81 | 0.69 | 0.67 | 84.81% | 82.49% |
| person | 0.71 | 0.59 | 0.73 | 0.71 | **102.88%** | **119.47%** |
| hobbit | 0.68 | 0.55 | 0.69 | 0.64 | **101.70%** | **115.85%** |
| iphone | 0.70 | 0.57 | 0.71 | 0.72 | **102.42%** | **126.42%** |
| movie | 0.82 | 0.74 | 0.81 | 0.78 | 98.70% | **105.20%** |
| sanders | 0.76 | 0.75 | 0.61 | 0.57 | 80.47% | 75.63% |
| Narr-KDML-2012 | 0.83 | 0.83 | 0.66 | 0.64 | 79.00% | 77.47% |
| archeage | 0.82 | 0.81 | 0.58 | 0.54 | 70.02% | 66.54% |
| SemEval18 | 0.77 | 0.77 | 0.63 | 0.59 | 81.31% | 77.35% |
| debate08 | 0.76 | 0.72 | 0.64 | 0.65 | 85.35% | 89.81% |
| HCR | 0.72 | 0.60 | 0.73 | 0.62 | **101.31%** | **103.84%** |
| STS-gold | 0.78 | 0.75 | 0.80 | 0.80 | **102.69%** | **107.34%** |
| SentiStrength | 0.75 | 0.74 | 0.71 | 0.67 | 94.35% | 89.85% |
| Target-dependent | 0.80 | 0.80 | 0.66 | 0.65 | 82.82% | 81.64% |
| VADER | 0.83 | 0.81 | 0.81 | 0.81 | 98.02% | **100.06%** |
| SemEval13 | 0.77 | 0.71 | 0.77 | 0.73 | **101.07%** | **102.83%** |
| SemEval17-test | 0.85 | 0.85 | 0.62 | 0.60 | 72.47% | 71.09% |
| SemEval16 | 0.82 | 0.81 | 0.80 | 0.77 | 96.97% | 95.57% |
| | | | | Average | 92.26% | 97.43% |
| | | | | St.dev. | 11.30% | 18.59% |

Table II. Classifiers accuracies and $F_{AVG}$ according to target-dataset model and closest CS1 model and its respectives ratios.

The experiment reported in Table II for CS1 was reproduced for all the distance metrics, and averages and standard deviations presented in the last two rows were summarized in Table III. Table III shows averages accuracy ratio and $F_{\text{AVG}}$ ratio on second and fourth columns, respectively, and

| Metric | Accuracy ratio average | Accuracy ratio st.dev. | $F_{AVG}$ ratio average | $F_{AVG}$ ratio st.dev. |
|---|---|---|---|---|
| ED1 | **92.72**% | 14.08% | 95.35% | 25.56% |
| ED2 | 87.45% | 19.44% | 89.76% | 30.92% |
| CS1 | 92.26% | **11.30**% | **97.43**% | **18.59**% |
| CS2 | 90.70% | 13.76% | 93.72% | 25.25% |
| CS3 | 89.62% | 14.38% | 92.26% | 27.44% |
| JD | 87.94% | 15.49% | 87.28% | 18.76% |
| RWMD | 87.82% | 12.09% | 88.18% | 20.12% |

Table III. Averages and standard deviations for accuracy and $F_{AVG}$ ratios according to metrics.

their standard deviations on third and fifth columns. The best results are presented in boldface type. As we can observe, CS1 achieved the best overall results in terms of $F_{\text{AVG}}$ (97,43%) with the lowest standard deviation value (18,59%), when used to select a source dataset to train a classifier. It means that, on average, the source dataset selected with this metric achieved 97,43% of the $F_{\text{AVG}}$ values obtained by classifiers trained with target datasets. In terms of accuracy, although CS1 achieved the second-best overall result (92,26%), its average performance is comparable to the best overall result achieved by ED1 (92,72%). This represents that CS1 achieved an average of 92,26% of the classification accuracy values when selecting the source dataset in comparison with the classifier trained with the target dataset itself. Nevertheless, CS1 presented the lowest standard deviation value (11,30%), which may indicate that it has a more consistent behavior in selecting a dataset to train a good classifier via transfer learning.

Next, in Tables IV and V, we present an "all versus all" comparison in terms of accuracy and $F_{AVG}$, respectively. Specifically, each cell in Tables IV and V shows the result achieved by applying the classifier trained on some source dataset (represented in the columns) to classify the instances from some target dataset (represented in the rows). The values in the main diagonal, i.e., the values in cells related to the same dataset in both row and column, refer to the the 10-fold cross-validation evaluation results on the dataset itself. For each target dataset, i.e., each row, the best results are presented in boldface type, and the top five results are underlined. Furthermore, "Top 1" and "Top 5" rows show the number of times each source dataset achieved the best and the top five best results, respectively. For each source dataset, the ratios between the results achieved by the classifier trained on it and the classifier trained on the dataset itself were calculated, and the average of those ratios are shown on "AVG % ratio" row.

| | iro | sar | ntu | S15 | stm | per | hob | iph | mov | san | Nar | arc | S18 | deb | HCR | STS | SSt | Tar | VAD | S13 | S17 | S16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| iro | 0.66 | 0.66 | 0.37 | 0.66 | 0.55 | 0.34 | 0.37 | 0.34 | 0.34 | 0.71 | 0.51 | 0.62 | 0.68 | 0.66 | 0.66 | **0.72** | 0.62 | 0.69 | 0.54 | 0.55 | 0.68 | 0.68 |
| sar | 0.54 | 0.56 | 0.52 | 0.54 | 0.70 | 0.46 | 0.46 | 0.46 | 0.46 | 0.61 | 0.65 | 0.72 | 0.70 | 0.52 | 0.56 | 0.61 | 0.72 | 0.68 | 0.58 | 0.61 | **0.77** | 0.59 |
| ntu | 0.43 | 0.53 | 0.81 | 0.43 | 0.79 | 0.57 | 0.57 | 0.57 | 0.57 | 0.71 | 0.80 | 0.75 | 0.82 | 0.58 | 0.67 | 0.71 | **0.86** | 0.74 | 0.82 | 0.74 | 0.73 | 0.71 |
| S15 | **0.85** | 0.84 | 0.25 | **0.85** | 0.63 | 0.15 | 0.15 | 0.15 | 0.15 | 0.81 | 0.42 | 0.66 | 0.70 | 0.84 | 0.84 | 0.84 | 0.49 | 0.60 | 0.32 | 0.37 | 0.64 | 0.45 |
| stm | 0.49 | 0.54 | 0.68 | 0.49 | 0.81 | 0.51 | 0.51 | 0.51 | 0.51 | 0.72 | 0.74 | 0.71 | 0.77 | 0.64 | 0.61 | 0.66 | **0.82** | 0.79 | 0.77 | 0.69 | 0.79 | 0.74 |
| per | 0.29 | 0.32 | 0.72 | 0.29 | 0.72 | 0.71 | 0.71 | 0.71 | 0.71 | 0.61 | 0.74 | 0.67 | 0.69 | 0.40 | 0.40 | 0.57 | 0.73 | **0.77** | 0.75 | 0.72 | 0.74 | 0.77 |
| hob | 0.32 | 0.38 | 0.71 | 0.32 | 0.67 | 0.68 | 0.68 | 0.68 | 0.68 | 0.45 | 0.70 | 0.65 | 0.52 | 0.36 | 0.37 | 0.39 | 0.70 | 0.69 | **0.72** | 0.69 | 0.70 | 0.69 |
| iph | 0.30 | 0.40 | 0.67 | 0.30 | 0.65 | 0.70 | 0.70 | 0.70 | 0.70 | 0.53 | 0.72 | 0.63 | 0.62 | 0.42 | 0.39 | 0.42 | 0.71 | 0.74 | 0.74 | 0.73 | 0.73 | **0.75** |
| mov | 0.18 | 0.23 | 0.81 | 0.18 | 0.75 | 0.82 | 0.82 | 0.82 | 0.82 | 0.43 | 0.78 | 0.70 | 0.58 | 0.24 | 0.32 | 0.38 | 0.76 | 0.78 | **0.84** | 0.81 | 0.76 | 0.83 |
| san | 0.53 | 0.56 | 0.50 | 0.53 | 0.69 | 0.47 | 0.47 | 0.47 | 0.47 | **0.76** | 0.63 | 0.62 | 0.75 | 0.68 | 0.59 | 0.65 | 0.65 | 0.69 | 0.61 | 0.62 | 0.75 | 0.65 |
| Nar | 0.40 | 0.49 | 0.77 | 0.40 | 0.81 | 0.60 | 0.60 | 0.60 | 0.60 | 0.73 | 0.83 | 0.76 | 0.82 | 0.55 | 0.62 | 0.66 | **0.85** | 0.84 | 0.82 | 0.80 | 0.84 | 0.81 |
| arc | 0.58 | 0.62 | 0.47 | 0.58 | 0.71 | 0.42 | 0.44 | 0.42 | 0.42 | 0.69 | 0.59 | **0.82** | 0.73 | 0.64 | 0.64 | 0.68 | 0.65 | 0.67 | 0.58 | 0.58 | 0.78 | 0.62 |
| S18 | 0.53 | 0.58 | 0.55 | 0.53 | 0.69 | 0.47 | 0.47 | 0.47 | 0.47 | 0.69 | 0.67 | 0.69 | **0.77** | 0.61 | 0.61 | 0.64 | 0.72 | 0.75 | 0.63 | 0.66 | 0.75 | 0.71 |
| deb | 0.63 | 0.64 | 0.43 | 0.63 | 0.66 | 0.37 | 0.37 | 0.38 | 0.37 | 0.67 | 0.64 | 0.64 | 0.67 | **0.76** | 0.64 | 0.66 | 0.69 | 0.69 | 0.61 | 0.64 | 0.67 | 0.67 |
| HCR | 0.72 | 0.72 | 0.31 | 0.72 | 0.67 | 0.29 | 0.38 | 0.30 | 0.28 | 0.73 | 0.51 | 0.72 | **0.73** | 0.65 | 0.72 | 0.73 | 0.66 | 0.73 | 0.66 | 0.65 | 0.73 | 0.73 |
| STS | 0.69 | 0.70 | 0.56 | 0.69 | 0.74 | 0.31 | 0.31 | 0.31 | 0.31 | 0.79 | 0.63 | 0.67 | **0.80** | 0.73 | 0.73 | 0.78 | 0.71 | 0.60 | 0.56 | 0.54 | 0.67 | 0.51 |
| SSt | 0.41 | 0.46 | 0.67 | 0.41 | 0.72 | 0.59 | 0.59 | 0.59 | 0.59 | 0.62 | 0.72 | 0.65 | 0.71 | 0.51 | 0.53 | 0.60 | **0.75** | 0.72 | 0.71 | 0.71 | 0.71 | 0.71 |
| Tar | 0.50 | 0.51 | 0.53 | 0.50 | 0.65 | 0.50 | 0.51 | 0.50 | 0.50 | 0.62 | 0.66 | 0.64 | 0.69 | 0.53 | 0.54 | 0.60 | 0.70 | **0.80** | 0.68 | 0.66 | 0.76 | 0.72 |
| VAD | 0.31 | 0.40 | 0.76 | 0.31 | 0.73 | 0.69 | 0.70 | 0.69 | 0.69 | 0.59 | 0.79 | 0.65 | 0.66 | 0.47 | 0.50 | 0.56 | 0.81 | 0.79 | **0.83** | 0.80 | 0.73 | 0.80 |
| S13 | 0.27 | 0.31 | 0.75 | 0.27 | 0.75 | 0.73 | 0.73 | 0.73 | 0.73 | 0.54 | 0.77 | 0.73 | 0.67 | 0.41 | 0.45 | 0.48 | **0.81** | 0.79 | 0.77 | 0.77 | 0.78 | 0.77 |
| S17 | 0.63 | 0.64 | 0.40 | 0.63 | 0.69 | 0.37 | 0.41 | 0.38 | 0.37 | 0.77 | 0.57 | 0.78 | 0.81 | 0.70 | 0.67 | 0.73 | 0.69 | **0.85** | 0.62 | 0.66 | 0.85 | 0.80 |
| S16 | 0.27 | 0.29 | 0.73 | 0.27 | 0.74 | 0.73 | 0.73 | 0.73 | 0.73 | 0.48 | 0.77 | 0.67 | 0.60 | 0.38 | 0.38 | 0.42 | 0.80 | 0.80 | 0.79 | 0.80 | 0.78 | **0.82** |
| Top 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 3 | 1 | 0 | 1 | **5** | 3 | 3 | 0 | 1 | 2 |
| Top 5 | 1 | 1 | 2 | 1 | 6 | 1 | 1 | 1 | 1 | 5 | 6 | 3 | 12 | 3 | 1 | 3 | **12** | **15** | 9 | 3 | **14** | 13 |
| AVG % ratio | 63% | 68% | 77% | 63% | 93% | 69% | 70% | 69% | 69% | 85% | 89% | 90% | 93% | 73% | 74% | 81% | **95%** | **97%** | 89% | 88% | **98%** | 93% |

Table IV. Accuracies for models trained with columns datasets applied to target datasets (rows).

In terms of accuracy (Table IV), we can observe that datasets Target-dependent (Tar column) and SemEval17-test (S17 column) achieved the best overall results regarding their use as source datasets

| | iro | sar | ntu | S15 | stm | per | hob | iph | mov | san | Nar | arc | S18 | deb | HCR | STS | SSt | Tar | VAD | S13 | S17 | S16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| iro | 0.53 | 0.53 | 0.28 | 0.53 | 0.57 | 0.17 | 0.23 | 0.17 | 0.17 | 0.65 | 0.51 | 0.56 | 0.63 | 0.59 | 0.55 | 0.65 | 0.63 | 0.68 | 0.54 | 0.56 | 0.64 | **0.68** |
| sar | 0.37 | 0.43 | 0.41 | 0.37 | 0.70 | 0.30 | 0.30 | 0.30 | 0.30 | 0.55 | 0.63 | 0.72 | 0.69 | 0.37 | 0.47 | 0.53 | 0.71 | 0.68 | 0.53 | 0.56 | **0.77** | 0.58 |
| ntu | 0.26 | 0.46 | 0.80 | 0.26 | 0.79 | 0.42 | 0.42 | 0.42 | 0.42 | 0.70 | 0.79 | 0.75 | 0.82 | 0.53 | 0.66 | 0.70 | **0.86** | 0.73 | 0.81 | 0.71 | 0.72 | 0.67 |
| S15 | 0.79 | 0.80 | 0.24 | 0.79 | 0.68 | 0.04 | 0.04 | 0.04 | 0.04 | 0.78 | 0.48 | 0.71 | 0.74 | 0.79 | **0.80** | 0.79 | 0.55 | 0.66 | 0.35 | 0.42 | 0.70 | 0.51 |
| stm | 0.33 | 0.44 | 0.65 | 0.33 | 0.81 | 0.34 | 0.36 | 0.34 | 0.34 | 0.71 | 0.74 | 0.71 | 0.76 | 0.58 | 0.57 | 0.61 | **0.82** | 0.79 | 0.77 | 0.67 | 0.79 | 0.73 |
| per | 0.13 | 0.20 | 0.61 | 0.13 | 0.69 | 0.59 | 0.59 | 0.59 | 0.59 | 0.62 | 0.69 | 0.69 | 0.71 | 0.37 | 0.34 | 0.58 | 0.71 | **0.78** | 0.70 | 0.67 | 0.75 | 0.75 |
| hob | 0.16 | 0.29 | 0.64 | 0.16 | 0.67 | 0.55 | 0.55 | 0.55 | 0.55 | 0.41 | 0.67 | 0.66 | 0.52 | 0.23 | 0.26 | 0.33 | **0.70** | 0.68 | 0.66 | 0.64 | 0.69 | 0.65 |
| iph | 0.14 | 0.32 | 0.64 | 0.14 | 0.66 | 0.57 | 0.59 | 0.57 | 0.57 | 0.52 | 0.72 | 0.64 | 0.63 | 0.36 | 0.30 | 0.36 | 0.72 | 0.74 | 0.72 | 0.73 | 0.74 | **0.75** |
| mov | 0.05 | 0.15 | 0.76 | 0.05 | 0.76 | 0.74 | 0.74 | 0.74 | 0.74 | 0.46 | 0.77 | 0.73 | 0.62 | 0.18 | 0.31 | 0.39 | 0.77 | 0.79 | **0.81** | 0.78 | 0.78 | 0.80 |
| san | 0.37 | 0.44 | 0.37 | 0.37 | 0.69 | 0.30 | 0.33 | 0.30 | 0.30 | **0.75** | 0.61 | 0.57 | 0.75 | 0.67 | 0.49 | 0.60 | 0.64 | 0.69 | 0.57 | 0.59 | 0.75 | 0.63 |
| Nar | 0.23 | 0.43 | 0.75 | 0.23 | 0.81 | 0.46 | 0.45 | 0.45 | 0.45 | 0.73 | 0.83 | 0.76 | 0.83 | 0.50 | 0.61 | 0.64 | **0.85** | 0.84 | 0.81 | 0.78 | 0.83 | 0.80 |
| arc | 0.42 | 0.51 | 0.37 | 0.42 | 0.71 | 0.25 | 0.29 | 0.25 | 0.25 | 0.66 | 0.57 | **0.81** | 0.71 | 0.60 | 0.56 | 0.63 | 0.65 | 0.67 | 0.54 | 0.56 | 0.78 | 0.61 |
| S18 | 0.37 | 0.48 | 0.47 | 0.37 | 0.69 | 0.30 | 0.31 | 0.30 | 0.30 | 0.65 | 0.66 | 0.68 | **0.77** | 0.53 | 0.53 | 0.59 | 0.72 | 0.75 | 0.59 | 0.64 | 0.75 | 0.70 |
| deb | 0.48 | 0.52 | 0.32 | 0.48 | 0.65 | 0.20 | 0.22 | 0.23 | 0.20 | 0.58 | 0.64 | 0.57 | 0.60 | **0.72** | 0.52 | 0.56 | 0.68 | 0.65 | 0.61 | 0.65 | 0.60 | 0.66 |
| HCR | 0.60 | 0.60 | 0.21 | 0.60 | 0.65 | 0.14 | 0.34 | 0.16 | 0.12 | 0.63 | 0.52 | 0.62 | 0.64 | 0.63 | 0.60 | 0.63 | 0.66 | 0.64 | 0.67 | 0.67 | 0.62 | **0.69** |
| STS | 0.56 | 0.62 | 0.55 | 0.56 | 0.75 | 0.15 | 0.15 | 0.15 | 0.15 | 0.77 | 0.64 | 0.68 | **0.80** | 0.66 | 0.70 | 0.75 | 0.72 | 0.61 | 0.55 | 0.52 | 0.68 | 0.49 |
| SSt | 0.24 | 0.36 | 0.61 | 0.24 | 0.72 | 0.43 | 0.45 | 0.43 | 0.43 | 0.59 | 0.71 | 0.65 | 0.70 | 0.44 | 0.47 | 0.57 | **0.74** | 0.73 | 0.67 | 0.68 | 0.71 | 0.69 |
| Tar | 0.33 | 0.37 | 0.43 | 0.33 | 0.64 | 0.33 | 0.36 | 0.34 | 0.33 | 0.56 | 0.65 | 0.63 | 0.67 | 0.42 | 0.43 | 0.54 | 0.70 | **0.80** | 0.65 | 0.64 | 0.75 | 0.71 |
| VAD | 0.15 | 0.33 | 0.71 | 0.15 | 0.74 | 0.57 | 0.59 | 0.57 | 0.56 | 0.59 | 0.78 | 0.66 | 0.67 | 0.43 | 0.48 | 0.55 | **0.81** | 0.79 | 0.81 | 0.78 | 0.74 | 0.79 |
| S13 | 0.12 | 0.19 | 0.68 | 0.12 | 0.75 | 0.61 | 0.62 | 0.61 | 0.61 | 0.54 | 0.73 | 0.74 | 0.68 | 0.36 | 0.43 | 0.47 | **0.80** | 0.79 | 0.71 | 0.71 | 0.78 | 0.73 |
| S17 | 0.48 | 0.52 | 0.26 | 0.48 | 0.70 | 0.20 | 0.29 | 0.22 | 0.20 | 0.75 | 0.55 | 0.77 | 0.80 | 0.66 | 0.57 | 0.69 | 0.69 | **0.85** | 0.60 | 0.65 | 0.85 | 0.80 |
| S16 | 0.12 | 0.15 | 0.65 | 0.12 | 0.75 | 0.61 | 0.62 | 0.61 | 0.61 | 0.47 | 0.74 | 0.69 | 0.61 | 0.32 | 0.31 | 0.38 | 0.80 | **0.81** | 0.74 | 0.77 | 0.79 | 0.81 |
| Top 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 1 | 1 | 0 | **7** | **4** | 1 | 0 | 1 | 3 |
| Top 5 | 1 | 1 | 1 | 1 | 10 | 0 | 0 | 0 | 0 | 3 | 5 | 4 | 10 | 2 | 1 | 3 | **16** | **16** | 5 | 5 | **16** | 11 |
| AVG % ratio | 47% | 59% | 74% | 47% | 101% | 54% | 58% | 55% | 54% | 89% | 95% | 98% | 100% | 70% | 71% | 81% | **104%** | **105%** | 94% | 94% | **106%** | 99% |

Table V. $F_{AVG}$ for models trained with columns datasets applied to target datasets (rows).

in the classification via transfer learning. While the dataset Target-dependent achieved the top five best results in 15 out of the 22 datasets (97% of AVG % ratio), dataset SemEval17-test achieved the top five best results in 14 out of the 22 datasets (98% of AVG % ratio). It is worth mentioning that dataset SentiStrength (SSt column) achieved the best overall results in five out of the 22 datasets, and the top five best results in 12 out of the 22 datasets (95% of AVG % ratio). These ratios indicate the average gain of classification accuracy achieved by the source dataset in one column compared to the classifier's accuracy results trained with the target dataset itself. That means they had almost the same accuracy of the target dataset classifier, which is quite remarkable.

Similarly, in terms of $F_{\text{AVG}}$ (Table V), we can notice that datasets SemEval17-test, Target-dependent, and SentiStrength also achieved the best overall results. Their AVG % ratios for the $F_{\text{AVG}}$, respectively 106%, 105%, and 104%, outperformed the results obtained using the classifiers trained with the target datasets themselves. Interestingly, these three datasets are among the ones with the greatest number of tweets, which could indicate why they had such a good performance, independently of the distance to the target dataset. Moreover, the variety in SemEval17-test and Target-dependent subjects, respectively entities, products, and events, and celebrities, products, and companies, may help to explain their performance.

## 6. CONCLUSIONS AND FUTURE WORK

This article intended to determine the most suitable distance metric between two datasets to choose a labeled dataset to train a target classifier via transfer learning. For this purpose, we evaluated four types of metrics in a large set of 22 Twitter datasets in English, achieving promising results.

In fact, one particular combination of distance metric and dataset representation reached a notorious performance over the seven combinations employed: the cosine similarity applied to the datasets represented with BoW and TF-IDF (CS1). This metric achieved the best results in term of $F_{AVG}$ and the second best in terms of accuracy. In terms of accuracy, the best metric was ED1, although that value was very close to CS1's accuracy. Moreover, the CS1 metric presented the smallest standard deviations, showing that it has a more consistent behavior in predicting the target dataset's classes. This result reveals that selecting CS1 as the distance metric to choose a training dataset tends to reach good results in most of the cases.

Furthermore, the experiment conducted to verify if some dataset, independently of a distance metric, could be selected to build a proper performance classifier revealed that some of the datasets reach good generalization. SemEval17-test, SentiStrength, and Target-dependent had good results in terms of both accuracy and $F_{AVG}$. On average, they displayed a greater $F_{AVG}$ value than the classifier

trained by the target dataset itself.

Future work could use more distance metrics or change the datasets representation form to establish a closer relationship between a distance metric and performance metrics. Also, identifying which characteristics of those datasets lead to the best performance is a promising path for future investigation. It can start by extracting features from these datasets, like their dimension, or the vocabulary size. In addition, identifying when to rely on the distance metric or when to adopt the dataset with the best overall performance is a promising venue for future work.

## ACKNOWLEDGMENT

The authors thank the agencies CNPq and FAPERJ for the financial support.

## REFERENCES

Bravo-Marquez, F., Frank, E., Mohammad, S. M., and Pfahringer, B. Determining word-emotion associations from tweets by multi-label classification. In *Proceedings of the 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, Omaha, USA, pp. 536–539, 2016.

Cambria, E., Poria, S., Gelbukh, A., and Thelwall, M. Sentiment analysis is a big suitcase. *IEEE Intelligent Systems* 32 (6): 74–80, 2017.

Carvalho, J. and Plastino, A. On the combination and evaluation of state-of-the-art features in twitter sentiment analysis. *Artificial Intelligence Review*, 2020.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*. Association for Computational Linguistics, Minneapolis, MN, 2019.

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. From word embeddings to document distances. In *Proceedings of the International Conference on Machine Learning*. PMLR, Lille, France, pp. 957–966, 2015.

Liu, B. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, San Rafael, USA, 2012.

Martínez-Cámara, E., Martín-Valdivia, M., López, L., and Montejo-Ráez, A. Sentiment analysis in twitter. *Natural Language Engineering* vol. 20, pp. 1–28, 01, 2014.

Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. Efficient Estimation of Word Representations in Vector Space. *CoRR* vol. abs/1301.3781, 2013.

Pan, S. J. and Yang, Q. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering* 22 (10): 1345–1359, 2010.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* vol. 12, pp. 2825–2830, 2011.

Petrovic, S., Osborne, M., and Lavrenko, V. The edinburgh twitter corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*. Association for Computational Linguistics, Los Angeles, CA, pp. 25–26, 2010.

Plank, B. and van Noord, G. Effective measures of domain similarity for parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Portland, USA, pp. 1566–1576, 2011.

Remus, R. Domain adaptation using domain similarity- and domain complexity-based instance selection for cross-domain sentiment analysis. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining Workshops*. IEEE, Brussels, Belgium, pp. 717–723, 2012.

Ruder, S. *Neural transfer learning for natural language processing*. Ph.D. thesis, NUI Galway, 2019.

Ruder, S. and Plank, B. Learning to select data for transfer learning with Bayesian Optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, pp. 372–382, 2017.

Santos, J. S., Paes, A., and Bernardini, F. Combining labeled datasets for sentiment analysis from different domains based on dataset similarity to predict electors sentiment. In *Proceedings of the 2019 8th Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, Salvador, Brazil, pp. 455–460, 2019.

Van Asch, V. and Daelemans, W. Using Domain Similarity for Performance Estimation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*. Association for Computational Linguistics, Uppsala, Sweden, pp. 31–36, 2010.