Semi-Supervised Sentiment Analysis of Portuguese Tweets with Random Walk in Feature Sample Networks

Pedro Gengo[1] and Filipe A. N. Verri[2]

[1] Data Science Team, Itaú Unibanco, São Paulo, SP, Brazil
pedro.gengo.lourenco@gmail.com
[2] Computer Science Division, Aeronautics Institute of Technology (ITA), São José dos Campos, SP, Brazil
verri@ita.br

Abstract. Nowadays, a huge amount of data is generated daily around the world and many machine learning tasks require labeled data, which sometimes is not available. Manual labeling such amount of data may consume a lot of time and resources. One way to overcome this limitation is to learn from both labeled and unlabeled data, which is known as semi-supervised learning. In this paper, we use a positive-unlabeled (PU) learning technique called Random Walk in Feature-Sample Networks (RWFSN) to perform semi-supervised sentiment analysis, which is an important machine learning that can be achieved by classifying the polarity of texts, in Brazilian Portuguese tweets. Although RWFSN reaches excellent performance in many PU learning problems, it has two major limitations when applied in our problem: it assumes that samples are long texts (many features) and that the class prior probabilities are known. We leverage the technique by augmenting the data representation in the feature space and by adding a validation set to better estimate the class priors. As a result, we identified unlabeled samples of the positive class with precision around at 70% in higher labeled ratio, but with high standard deviation, showing the impact of data variance in results. Moreover, given the properties of the RWFSN method, we provide interpretability of the results by pointing out the most relevant features of the task.

Keywords: Sentiment analysis · Semi-supervised classification · Positive-unlabeled learning · Random walk.

1 Introduction

Traditionally, machine learning tasks are divided in two categories: supervised learning, tasks whose input data are labeled, or unsupervised learning, when data are unlabeled. However, a third paradigm, called semi-supervised learning, combines labeled and unlabeled data and take advantage of this combination [13]. The study of techniques in this paradigm is extremely important because data

has been generated at an increasing rate and, in many applications, manual labelling is expensive and time-consuming.

One particular task is sentiment analysis, which is also called opinion mining. The goal of sentiment analysis is to extract sentiments and opinions from natural language text using computational methods [7]. A more specific task is polarity classification, which consists of classifying texts in the following classes: positive, negative and neutral. Methods used in this task range from machine learning algorithms to lexical or distant approach [3–5, 10, 11].

We model the task of polarity classification as a problem where we have few labeled examples of the positive class and all others are unlabeled. This problem is called positive-unlabeled (PU) learning, and is an inner class problem of semi-supervised learning. The goal of PU learning is to label all unlabeled input samples at once (transductive learning) or to construct a function that discriminates positive and negative samples (inductive learning) [9].

One technique of PU learning is Random Walk in Feature-Sample Networks (RWFSN) [12], which is a graph-based technique with steps: *a)* Convert the dataset into a sparse binary representation. *b)* Create a bipartite graph where samples and features are the vertices, and edges are the connection between a sample and a feature. *c)* Perform a random walk process over the graph, applying a scaling factor(constant) at labeled samples. *d)* Use the limiting distribution of the Markov chain to calculate the positive-class confidence of unlabeled sample. *e)* Order the unlabeled samples by their positive-class confidence and, with knowledge of the positive-class prior probability, classify the unlabeled samples.

A limitation with this technique, showed at [12], is that the classification step depends on the assumption of knowing the positive-class prior probability and this information may not be known in real-world problems.

In this paper, we adapted a dataset of Brazilian Portuguese tweets to be able to use it in the PU learning technique described to classify unlabeled samples. We also proposed a modification of the technique, using a validation set to choose the threshold of positive-class confidence without the need of knowing the prior probabilities.

The rest of this paper is organized as follows. In Section 2, we present the proposed modification of the technique to deal with the problem of unknown prior probability. Section 3 describes the preprocessing steps used to treat tweets of the dataset. Finally, Sections 4 and 5 show our results and conclude this paper.

2 Model Description

The general idea of the model proposed in [12] is that it receives a dataset where each sample is either a positive or negative sample and only a few positive samples are labeled. Using the prior probability P^+ and the positive-class confidence, it classifies the unlabeled data.

In the following subsections, we explain the steps of the learning algorithm.

2.1 Construction of the Feature-Sample Network

The *Feature-Sample Network*, $\mathcal{G}$, is a bipartite complex network whose edges associates samples and features of the dataset $\mathcal{D}$. So, the vertices are samples and features, and an edge exists only where we have the presence of the feature in a sample.

We can construct this complex network defining the vertex set $\mathcal{V}$ like $\{v_1, ..., v_N, v_{N+1}, ..., v_{N+M}\}$, where N represents the number of samples and M the number of features in the dataset. Besides that, an edge exists only between samples and features. So, we can define the adjacency matrix of this graph as:

$$A = \begin{bmatrix} 0 & X \\ X^T & 0 \end{bmatrix} \text{ ,where X is the dataset.}$$

An import condition is that $\mathcal{G}$ has a single connected component. If it is false, only the largest connected component will be considered.

2.2 Modeling of the Random Walk Process

We want to reach the stationary distribution $\vec{\pi}$ for an irreducible Markov Chain. So, we model the transition matrix P to guarantee the existence and uniqueness of this limiting distribution.

The limiting distribution of a random walk is reached independently of the initial conditions if the Markov chain is ergodic. To satisfy this requirement, we model the transition matrix P as

$$p_{ij} = \frac{w_{ij}\nu_j}{\lambda\nu_i},$$

where the matrix $W = (w_{ij})$ has elements

$$w_{ij} = \begin{cases} 1, & \text{if i = j} \\ \beta a_{ij}, & \text{if } x_i \text{ is a positive sample} \\ a_{ij}, & \text{otherwise,} \end{cases}$$

$\vec{\nu}$ is the eigenvector associated with the leading eigenvalue λ of the matrix W and β is a hyperparameter called scaling factor, and a_{ij} is the element of the adjacency matrix of the bipartite graph.

2.3 Estimation of the Positive-class confidence

The transition matrix P describes a system with a limiting distribution $\vec{\pi}$ reached independently of the initial setting. We expected that the limiting probabilities related to positive samples are greater than the ones associated with negative samples.

So, we estimate the positive-class confidence $f(\vec{x_i}) = \pi_i$ for all unlabeled samples. The stochastic vector $\vec{\pi} = (\pi_i)$ is a eigenvector associated with the leading eigenvalue of matrix P^T.

Algorithm 1 illustrates the steps of RWFSN method [3].

[3] Available on https://github.com/pedrogengo/RWFSN

Algorithm 1: RWFSN

Data: Dataset with samples as binary feature vectors, β
Result: Positive-class confidence of unlabeled data ($f(\vec{x_i})$)
Initialization;
Create the adjacency matrix A;
Remove all disconnected elements;
Create the matrix $W = (w_{ij})$;
Do the spectral decomposition of the matrix W;
$\lambda \leftarrow$ leading eigenvalue of matrix W;
$\vec{\nu} \leftarrow$ eigenvector associated with λ;
Calculate the transition matrix P;
Do the spectral decomposition of the matrix P^T;
$\vec{\pi} \leftarrow$ eigenvector associated with the leading eigenvalue of P^T;
$f(\vec{x_i}) \leftarrow \vec{\pi}$;
end

2.4 Classification of the unlabeled samples

Differently from the original model, where the classification of unlabeled samples is made by using the prior probability of positive class to determine the threshold to classify as either positive or negative, we propose a new way to perform this classification.

Based on the idea of validation set, widely used in supervised learning, we propose to split a fraction of the labeled data, once it was shown that the model works well with few labeled samples, and use this split as an unlabeled data. So, after run the Algorithm 1, we could choose the threshold ($f^n_{validation}$) based on this samples that we know and were not scaled by β remaining on the same scale as unlabeled data.

The predicted class $c(\vec{x_i})$ of an unlabeled sample $\vec{x_i}$ regarding the validation data is

$$c(\vec{x_i}) = \begin{cases} +1, & \text{if } f(\vec{x_i}) > f^n_{\text{validation}} \\ -1, & \text{otherwise.} \end{cases}$$

In Fig. 1, we observe the proposed method to select the threshold to classify the unlabeled data.

3 Dataset and preprocessing

To perform sentiment analysis of Portuguese tweets, the TweetSentBR dataset [2] was used, which contains 15047 tweets ID's and their classifications. The tweets were related to Brazilian program shows. Of the total, 44% are positive, 26% are neutral and 29% negatives. However, during the scrapping using Twitter's API and the tweet's ID as keys, some of tweets were not found. Therefore, the collected dataset is formed by 11610 samples, where 45% are positive, 25% are neutral and 30% negative.

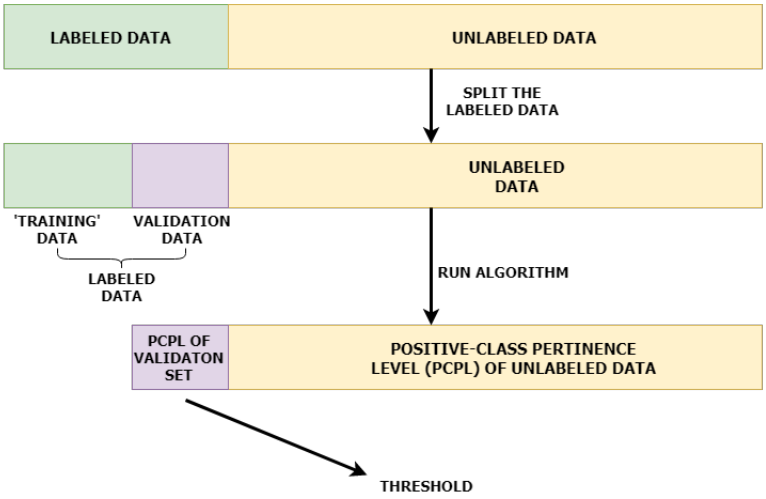

Fig. 1. Proposed change to use a validation set to choose better the threshold of positive-class confidence. We split the labeled data in two parts: training set and validation set. The validation set is treated as unlabeled data. Thus, we can choose the threshold based on positive-class confidence of validation set, which we known that are positive samples.

When we analyzed some of positive, neutral and negative tweets, we observed that, in some cases, the neutral tweets had a subtle difference from the other classes, as we can see in this neutral example, which has an positive emoji (heart):

Example: Consegui achar os ep de hoje do Master chef ♡ foi o ep 11 né gente ? #MasterChefBR

Thus, we decided to use only the positive and negative classes in our experiments once their difference is stronger. In Table 1, we can see the class distribution over the dataset used in the experiments.

Table 1. Class distributions of data used

Class	Frequency	Total
Positive	0.606	5232
Negative	0.394	3400
Total		**8632**

RWFSN technique requires that the input data satisfy 3 requirements, which are:

- Each sample is a binary feature vector;
- An attribute with value 1 indicates the presence of a characteristic of that data instance;
- The similarity of two samples depends only on the number of shared characteristics.

Tweets are short messages, restricted to 280 characters in length, but by the time the dataset was labeled, they were restricted to 140 characters. Because of this short length, people use acronyms, emoticons and other characters to substitute formal words [1]. The problem with that is we need to create a bipartite graph with the largest number of connections between positive class examples and with the use of acronyms, abbreviations, emoticons and others, similar tweets did not present shared characteristics.

So, to satisfy the requirements and solve the mentioned problem it was necessary to preprocess the dataset to achieve a binary representation with the largest numbers of connections (less sparse), relevant features and viable dimensionality.

The steps performed to tokenize and to preprocess the data were:

1. A binary feature which indicates the presence of an exclamation mark in the tweet (1) or not (0) was created. (flag_exclamation)
2. A binary feature which indicates the presence of a question mark in the tweet (1) or not (0) was created. (flag_question)
3. All characters were converted to lowercase.
4. HTML entities were replaced by their character and ASCII escape sequences were replaced by a blank space.

   Example: "\n" → " "
   "&lt" → "<"

5. Accents were removed.
6. Usernames (an @ symbol followed by up to 15 characters) were replaced by the tag __user__.
7. Hashtags (a # symbol followed by any sequence of characters) were replaced by the tag __hashtag__.
8. A binary feature which indicates the presence of words with sequence of repeated characters was created. (flag_n_letter)
9. A sequence of repeated characters was replaced by only one character. In this context of modeling the problem as a bipartite graph we wanted to have the largest number of connections. So, we would like to reduce this words with repeated characters to an unique form, in order to have fewer variations of the same word, resulting in more connections.

   Example: "ammooooooooo" → "amo"

10. Punctuation marks were removed.

11. A binary feature which indicates the presence of positive emojis in unicode format was created. To do so, we used a dictionary of emoji's sentiment [6]. So, we iterated the tweet and if an emoji was found we check its polarity magnitude. If this polarity was positive (greater than 0.1), this feature was 1, else 0 (flag_pos_emoji).
12. As in the previous topic, a binary feature which indicates the presence of negative emojis was created (flag_neg_emoji).
13. We performed a data augmentation by inserting root words or infinitive verb tense based on *thesaurus* Delaf Unitex [8] as the following procedures: creating a key-value structure to store the root words and their similar words or infinitive verb tense (we treated the words with repeated characters in this dictionary to follow the preprocessing done with tweets); inserting to this dictionary some acronyms and abbreviations frequently used, which were detected by tweet's analysis; ultimately, we iterated all tweets and inserted the root word or the infinitive verb tense of each word right after it.

The following tweet is used to demonstrate how was the output after preprocessing.

Original tweet:
Vou ficar tonta @pefabiodemelo #conversacombial #MasterChefBR
Preprocessed tweet:
vou ir ficar ficar tonta tonto __user__ __hashtag__ __hashtag__

After preprocessing the data, we tokenized the tweets and used bag of words (BoW) approach to represent the data as a binary vector. When we tokenized the data, we obtained 10986 different tokens. We selected only the 199 most frequent features because we achieved better results and reduced the sparsity of the matrix.

4 Results and Discussion

This section is divided in three subsections. In the first one, we compare the polarity classification of Portuguese tweets using RWFSN and a baseline model based on the neighborhood graph. In the second one we present the results using the proposed method (validation set) to choose the threshold. And in the last one we present a discussion about interpretability of the model and relevant features.

4.1 Performance Comparison of Polarity Classification

We compare the RWFSN model with a baseline semi-supervised method. Such baseline method is based on the neighborhood graph and we choose this method because it uses the same classification mechanism providing a fair comparison. We use the prior probability to perform the classification step in order to prove that the method can be extended to the task of polarity classification of tweets.

The baseline method is based on the construction of a k-NN graph of the dataset, where distance between two sample was done by using the Jaccard index. We calculated the positive-class confidence as

$$f^{k-NN}(\vec{x_i}) = (\min_j l_{ij})^{-1},$$

where l_{ij} is the length of the shortest path between vertices v_i and v_j. The positive class pertinence of a sample is inversely proportional to the shortest distance from the associated vertex to any labeled vertex.

To compare the performance between models, we observed the accuracy and the F-score, in order to compile precision and recall. We tested each ratio with 3 different samples and reported the average of metrics. The results are shown at the Table 2 with the respective best parameters.

Table 2. Comparison between the methods. the best parameter is shown.

Labeled Ratio	RWFSN		k-NN graph	
	Accuracy (β)	**F-Score**	**Accuracy(k)**	**F-Score**
1%	0.5203 (17)	0.6035	**0.5792 (30)**	**0.6376**
5%	**0.5819 (14)**	**0.65175**	0.5779 (30)	0.6373
10%	**0.6154 (9)**	**0.6759**	0.6045 (30)	0.6668

As we can see in Table 2, as the labeled sample ratio increases the RWFSN performance surpasses the baseline model. Although the difference was not large, this behavior suggests that RWFSN can exploit the label information more efficiently than the KNN graph.

4.2 Choice of threshold using Validation Set

We also perform tests using a validation set, which corresponds to 20% of the size of labeled set, to choose the threshold instead of using the prior probability, which is unknown in most of real-world problems. With the validation set we needed to choose a specific positive-class confidence of the samples in this set. So, in order to be conservative, we choose the higher positive-class confidence, because there is an idea that unlabeled samples that have confidence greater than the largest confidence of the validation set are probably also positive.

To evaluate the performance of this choice we used as metric the precision, which tell us how many examples were really positive of what we predict positive, and F-Score. We also performed a Receiver Operating Characteristic (ROC) Curve analysis using all the positive-class confidences of the validation set, in order to evaluate the performance with different choices of positive-class confidence in validation set.

We used different labeled ratio with the respective optimal β, which has been found empirically and we tested in 10 differents samples for each ratio.

The average of precision and F-Score and its standard deviation achieved are presented at Table 3 and the ROC Curves are presented at Fig. 2.

Table 3. Results using the higher positive-class confidence of the validation set for different labeled ratio

Labeled Ratio	Optimal β	Precision	F-Score
1%	17	0.63±0.02	0.12±0.10
5%	16	0.65±0.09	0.04±0.03
10%	9	0.70±0.04	0.01±0.01

As we can see in Table 3, we verified that the precision increases with more labeled data and the F-Score decreases. Another important result is about the impact of variance due to the sample used, which can be verified by the order of the standard deviation in precision. In the Fig. 2 we can corroborate what is verified at Table 3, as we show different curves that depend on the sample used.

4.3 Relevant Features

One important aspect of the RWFSN is that the states associated with each relevant features of the positive class has high stationary probabilities. So, if we look inside the positive-class confidence of features set, we can see which features are relevant to the positive class. With that, we can extend this technique to classify the polarity of words in a specific context or make a feature selection.

At Table 4, we can see the 10 most relevant features associated with the positive class using 5% of labeled samples and β equals to 16.

Table 4. Top 10 most relevant features using 5% of labeled samples and β equals to 16.

Position	Word
#1	flag_n_letter
#2	ser
#3	lindo
#4	do
#5	amor
#6	que
#7	flag_exclamation
#8	flah_pos_emoji
#9	de
#10	como

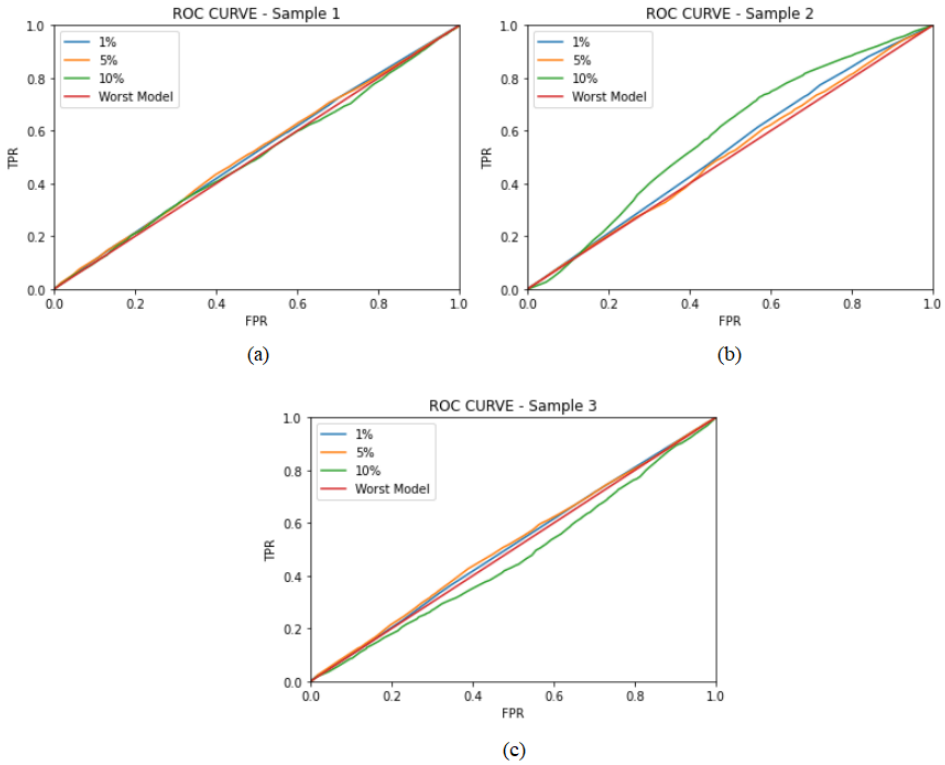

Fig. 2. ROC curves using different labeled ratio and different samples. Each curve color represents one different labeled ratio. As we can see in (a), (b) and (c) we obtained different results for each of the samples, showing the impact of the variance on the results.

Not surprisingly, the most relevant words for positive class, showed in Table 4, presents some of the features we believed that were important to the positive class classification. Some Portuguese stopwords appeared in this Table because we decided to keep them at preprocessing phase, once we wanted to achieve the largest number of connections at graph for the RWFSN model.

5 Conclusion and Future Work

The polarity classification task is very important in different scenarios such as product reviews and analysis of social media content. However, many machine learning methods suffers from the need of a large amount of labeled data during their training phases. In this paper, we proposed a preprocessing approach of the Portuguese tweets dataset to use this dataset in RWFSN, as well as a change in the classification mechanism of the model.

The results shown that model and classification mechanism proposed can be used to perform this task, but the issue of matrix sparsity should be further explored and it is important to take care of the sample used and about the distribution of your dataset, due to the impact of the data variance in the results. Moreover, this model has higher interpretability, which can be explored in other tasks such as word polarity detection. However, the limitation of the parameter β can impair the use of the model in this task.

As future works, we intend to explore ways to reduce the matrix sparsity by using sentiment lexicon for Portuguese and compare the presented model with traditional machine learning techniques. We also intend to explore a way to find the optimal β and use the model presented for active learning techniques.

Acknowledgment

This work was supported by Itaú-Unibanco.

Any opinions, findings, and conclusions expressed in this manuscript are those of the authors and do not necessarily reflect the views, official policy or position of Itaú-Unibanco.

References

1. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.: Sentiment analysis of twitter data. In: Proceedings of the Workshop on Languages in Social Media. pp. 30–38. LSM '11, Association for Computational Linguistics, Stroudsburg, PA, USA (2011), http://dl.acm.org/citation.cfm?id=2021109.2021114
2. Brum, H.B., das Graças Volpe Nunes, M.: Building a sentiment corpus of tweets in brazilian portuguese. CoRR **abs/1712.08917** (2017), http://arxiv.org/abs/1712.08917
3. Corrêa Jr, E.A., Marinho, V.Q., Santos, L.B.d., Bertaglia, T.F.C., Treviso, M.V., Brum, H.B.: Pelesent: Cross-domain polarity classification using distant supervision

4. Dos Santos, C., Gatti, M.: Deep convolutional neural networks for sentiment analysis of short texts. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 69–78 (2014)
5. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford **1**(12), 2009 (2009)
6. Kralj Novak, P., Smailović, J., Sluban, B., Mozetič, I.: Sentiment of emojis. PLoS ONE **10**(12), e0144296 (2015), http://dx.doi.org/10.1371/journal.pone.0144296
7. Liu, B.: Sentiment analysis and opinion mining, pp. 1–135. Cambridge University Press, New York, NY, USA (2015)
8. Muniz, M.C.M.: A construção de recursos lingüístico-computacionais para o português do brasil: o projeto de unitex-pb. São Carlos (2004)
9. Mũnoz-Marí, J., Bovolo, F., Gómez-Chova, L., Bruzzone, L., Camp-Valls, G.: Semisupervised one-class support vector machines for classification of remote sensing data. IEEE transactions on geoscience and remote sensing **48**(8), 3188–3197 (2010)
10. Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. In: LREc. vol. 10, pp. 1320–1326 (2010)
11. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Computational linguistics **37**(2), 267–307 (2011)
12. Verri, F.A.N., Zhao, L.: Random Walk in Feature – Sample Networks for Semi-Supervised Classification. In: 5th Brazilian Conference on Intelligent Systems Random. pp. 235–240 (2016). https://doi.org/10.1109/BRACIS.2016.41
13. Zhu, X., Goldberg, A.B.: Introduction to semi-supervised learning. Synthesis Lectures on Artificial Intelligence and Machine Learning **3**(1), 1–130 (2009). https://doi.org/10.2200/S00196ED1V01Y200906AIM006