Noname manuscript No.
(will be inserted by the editor)

Quantifying Emotions of Textual Utterances

Sofia Aparicio [1] · Bruno Martins [2]

Received: date / Accepted: date

Abstract Sentiment analysis has been one of the main application areas for Natural Language Processing (NLP) leveraging deep neural networks. Previous studies have covered the use of polarity and categorical sentiment classification, has well as approaches that only really on languages that have enough data to train a model. To the bets of our knowledge, that is still a gap when using dimensional sentiment analysis, specially in a multilingual domain, considering languages with few or none trying dataset. The main research goal in this paper is to understand what are the best models to quantify sentiment in a multilingual level. Several statistical and machine learning models where produced and compared in three different languages (English, Portuguese and Polish). This work shows promising results when inferring sentiment, even in languages other than English. The approaches can have several applications such as to monitor informal political online discussions and to lead to a better understanding of hate speech on social media. As well as, to better understand the mass opinion on trendy subjects.

Keywords Emotional ratings of text · Affective norms · Long Short-Term Memory · Recurrent Neural Networks · Machine learning.

1 Introduction

Sentiment analysis has been one of the main application areas for Natural Language Processing (NLP) leveraging deep neural networks. Sentiment analysis relies on the necessity to extract either negative and positive evaluation or estimate emotion. Human emotional ratings are nowadays frequently used within cognitive science, behavioural psychology and psycholinguistic research (33), social media analysis (17), among others.

Previous studies have covered the polarity (positive vs negative) of subjects (45). Two leading families of methods have been developed to represent human emotions (12). One is categorical, based on six universal basic emotions (BE) (11). The other is dimensional, advocating continuous numerical values that progress through multiple dimensions (43). Since it takes a significant amount of human resources to annotate words and textual utterances regarding sentiment, it was necessary to produce automatic methods to infer sentiment.

The first approaches to infer sentiment relied on song keyword extraction (27), considering the dimensions valence and arousal. However, this method does not take into consideration the structure of a sentence (24). Buechel et al.(6) wanted to foretell the emotion of a linguistic unit by a fine-grained analysis, using a regression instead of classification. Several studies were conducted using convolutional neural networks (CNN) (i.e. that considers the spatial organization of a sentence) and recurrent neural networks (RNN) (i.e. that consider the sequential organization) (7). Alswaidan et al.(2) work, three models were considered, gated recurrent unit followed by CuDNN concatenated with a CNN and a frequency inverse document frequency (TF-IDF), to better label the text according to emotional cate-

Sofia Aparicio
E-mail: asadacosta@gmail.com

Bruno Martins
E-mail: bruno.g.martins@ist.utl.pt

[1] Both with INESC-ID and Instituto Superior Tecnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001, Lisboa, Portugal.
[2] Both with INESC-ID and with the Computer Science and Engineering Department of IST-UL.

gories. Zahiri et al. (47) conducted emotion detection of the tv show Friends, through a sequence-based convolutional neural network (SCNN).

These works were mainly applied to the English language. So, to the best of my knowledge, there is still a gap when using deep learning to quantifying sentiment from languages with few or none training resources. It would also be fascinating to understand if there is a great difference between models that predict word-level sentiment and text-level. Understand what will perform better: CNN's or LSTM's. Pre-train a MLP layer and understand if it increases, or not, the performance of the models. Lastly, understand what model would perform better for this problem.

The methodology relied first on training MLP with word lexicons from six different languages. To infer the need to access all the syntactic structure to infer sentiment form a text, four models that do not take into consideration the syntactic structure and do not require training were created. MLP + Average, using the pre-trained MLP, calculates an average of the sentiment prediction of all the words to show the sentiment of the sentence. Average + MLP, similar to the previous one, but instead it is calculated a mean of the word embedding and after the MLP used the embeddings average to make a prediction. The last two models are more complex. The first considers windows of sizes between one and five words. Then, the average of all these pooling windows was calculated and to later apply the MLP. The last model suffered a little change since the MLP was applied after each pooling window and then calculated the average. And eight trainable models were conceived validated with two-fold cross-validation (i.e. LSTM, MLP+LSTM, CNN, MLP+CNN, CNN+MLP, Attention Concat, Attention Feacture Bassed, Attention Affine Transformation). The last three models were based on (28) work.

The results show that three trained models performed better (Attention Concat, Attention Feacture Bassed, Attention Affine Transformation); however, the Average word-level prediction model also showed promising results. LSTM tends to perform slightly better than CNN models. The difference was more evident in the arousal dimension. However, when the CNN and LSTM model were aligned with the pre-trained MLP, the results decreased, showing that a pre-trained MLP can decrease the performance of the model.

This article is organized as follows. Section 2 presents the state-of-the-art works in this field. Section3 describes two general approaches to extract sentiment from text. First, a statistical model is described, followed by Machine Learning models. Section 4 describes the obtained results, followed by Section 5, where we present our general conclusions and possible directions for future work.

2 Related Work

2.1 Assigning Emotion to Textual Utterances

In emotion analysis, word-level prediction differs a lot from assigning emotion values to larger linguistic units, such as paragraphs and sentences. (4) recognised three different approaches for emotion detection: keyword-based, learning-based, and hybrid. However, all these methods resort to different linguistic analysis tools (e.g., semantic level, sentence segmentation, parts of speech recognition, token level).

The first approach relies heavily on text preprocessing and relies on a domain specif theory, regarding several independent domains that hold different emotions. Thus, textual utterances are divided into words for the extraction of sentiment. (26) uses keyword spotting applied to a chat system to generate emotionally responsive messages. (27) make use of a keyword approach to analyse song verses, considering the valence and arousal space.

However, word-level problem solving cannot solve high-level linguistic prediction because of the way these words are combined(24). One example is a negation or irony, which can turn the meaning of the text, ignored if considering the words separately. The second, learning-based, consider a set of training data to shape a predictive model. This approach falls into two different categories depending on how the input is organised(7). One is arranged spatially, such as architectures that use convolutional neural networks (CNN). The other use sequential input data, typical for recurrent neural networks (RNN), long short term memory (LSTM) and general regression neural networks (GRNN). These two models will be explored in detail in the next subsection.

To apply these machine learning techniques, it is necessary to obtain a significant corpus, which is not possible in several cases. In the same way, it is challenging to solve these problems without using linguistic information. An interesting alternative is the last one, hybrid-based approach. In Alswaidan et al.(2) work, three models were considered, gated recurrent unit followed by CuDNN concatenated with a CNN and a frequency inverse document frequency (TF-IDF), to better label the text according to emotional categories.

2.2 Spatially and sequential Arquitectures

Firstly, considering the input arranged spatially, we have an early study conducted by (9). In this study, the primary goals were to evaluate the two models of sentiment representation, namely the dimensional and the categorical models, and determine what could be their applications and what could be the expected accuracy. For the categorical model, the text was converted into a VSM representation with TF-IDF weights. The VSM representation can then be reduced with LSA, probabilistic LSA (PLSA) and, Non-negative Matrix Factorization (NMF). These three translate the pseudo-documents into predefined categories. In the dimensional model, the authors resorted to ANEW and WordNet synsets. Each word is converted to the ANEW affective space. Afterwards, the words can be used to weight the sentence emotional place, naively. The NMF approach and dimensional model outperformed the other two.

A few years later, (6) wanted to foretell the emotion of a linguistic unit by a fine-grained analysis, using a regression model instead of classification and using two metrics to validate their results (Pearson correlation and root-mean-square error). The authors mapped the two emotion representations, translating the VAD output into a BE representation. Even though this method reduces performance, it still outperforms former systems that consider the three dimensions.

Since the amount of text documents rated in VA space is scarce, (34) chose to resort to two psychologically-trained annotators. Facebook posts were rated, firstly, considering the valence and arousal dimensions separately. Afterwards, the experts asked to rate the two aspects together. In sum, 2895 messages were evaluated and VA parameters were compared through age and gender of the writer, with the authors concluding that female post writers express both more arousal and valence. Later, a two linear regression model using a BoW representation, on 10-fold cross-validation with this data, reaches a high correlation to the annotated results, obtaining a Pearson correlation of 0.650 and 0.850 for valence and arousal, respectively.

With the limited research on the use of sequential input data and the need for more emotionally rated data, (47) started a new investigation. The dialogues from the show TV Friends were annotated considering seven emotions: sad, mad, scared, powerful, peaceful, joyful, and neutral. Since CNNs are not ideal for processing sequences and RNNs perform slowly, the authors induced four sequence-based convolution neural networks (SCNN). The input for all SCNN is the same: a matrix M, with dimensionality equal to the number of tokens in any utterance by the embedding size. Each row in M represents a token in the utterance.

3 Using Neural Word Embeddings for Extending Lexicons of Emotional Norms

We propose thirteen models, based on models described in the related work section. All the studies were conducted in six different languages: English, Spanish, Portuguese, Italian, and Polish.

In this section, we will start by describing the need for word embeddings, followed by an explanation of the models created in this study.

3.1 Word Embeddigs

Fist, it is necessary to consider that text can not be given directly to a machine learning algorithm; it is necessary to encode the text in a way intelligible for the algorithm. Word embeddings are vector representations for words, responsible for capturing there semantic or syntactic meaning. Several approaches were suggested over the years. Word2Vec(30), based on the skip-gram(29) model, is responsible for predicting the context of a given the word. However, the Word2Vec method does not allow the representation of words out of the vocabulary. FastText(19), based on the skip-gram model(29), proposes the use of word fragments to express word vectors allowing to represent words out of the vocabulary. In our model, we used FastText word vectors pre-trained on Common Crawl and Wikipedia, by Grave et all.(14). These embeddings are available in 157 different languages.

3.2 Models Exploring Statistics

We based this first part by a study that we conducted previously. In this study, we assign sentiment to words considering three dimensions, and we compared four different methods to predict the emotion of each word. One of the models that had a good performance was a simple multi-layer perceptron (MLP)(36), a set of neurons fully connected.

The MLP model was built through Keras[1], an open-source library integrated on top of TensorFlow, to allow building deep learning models. In this model, we have one input layer with the size of the embedding vector (vector of 300 doubles, in this particular case). Then a hidden state, created by Dense function, is composed of 100 neurons, plus one bias each. Where the

[1] https://github.com/keras-team/keras

weights (referred to as the kernel_initializer parameter) were initialized randomly and the biases with zeros. Also applied ReLu activation function to converge faster, be computationally cheaper and to allow a sparse activation. To decrease the chance of overfitting, we set a weight regularizer (kernel_regularizer parameter) as the L2 norm with the value 0.0001. The L2 norm, also known as the Euclidean norm, calculates the shortest distance between two points by summing the squared weights. On the output layer, we have a Dense layer with three neurons, one for each emotional dimension that we are considering. This layer has a linear activation function because of the continuous output values. This MLP was trained through 200 epochs, with a batch size of 64 and an Adam(22) optimizer.

We adapted the MLP from our previous work and trained it with datasets affective normas for words from six different languages: English(5; 39; 44), Spanish(35), Portuguese(40), Italian(31), German(37), and Polish(18). To all the datasets that are not English, they also have a column in English where the original text was translated. In those cases, when training the model, we considered both the word in English and on the original dataset language.

To observe the need for syntactic information when analyzing sentiment from the written test, we create four models with the MLP. These model do not take into consideration the syntactic structure of the textual utterances.

3.3 Models Exploring Machine learning

Yann LeCun, inspired by the human visual cortex, discovered by Hubel and Weisel(16), developed the Convolution and Polling architecture (25), also known as Convolutional Neural Networks (CNN). To do so, he mimicked three crucial features of the mammal brain: Local connections, to determine the way the neurons are related; Layering to define the hierarchy of elements that are learned; Scapial invariance to detect an object, disregarding, e.g. its standard size or orientation.

LeCun applied this technique to images, and it was years later that CNN was applied to NLP. The first studies were conducted by (10) in the area of semantic-role labelling, then by (20) and (21) in the field of sentiment analysis and question-type classification.

The main goal of CNN is to detect patterns across space, by firing when a determined pattern of words compared to a determined filter. CNN are composed of two layers, Convolution and Pooling.

Convolution Layer receives two inputs: a text translated into embeddings and a filter. The vector of embeddings is multiplied by the filter generating a Feature Map. Each filter takes into consideration a specific feature and can have differently sized, depending on what window size you want to consider. In our experiments, we regarded as sizes of word windows between one and five, as we show in Figure1. To reduce the Feature Maps, we pass them through a Pooling Layer, responsible for reducing the dimensionality, yet, recalling the important information. There are two types of Pooling operations, and we can see in the Figure that the one used in our models was Average Pooling, responsible for returning the average of all values.

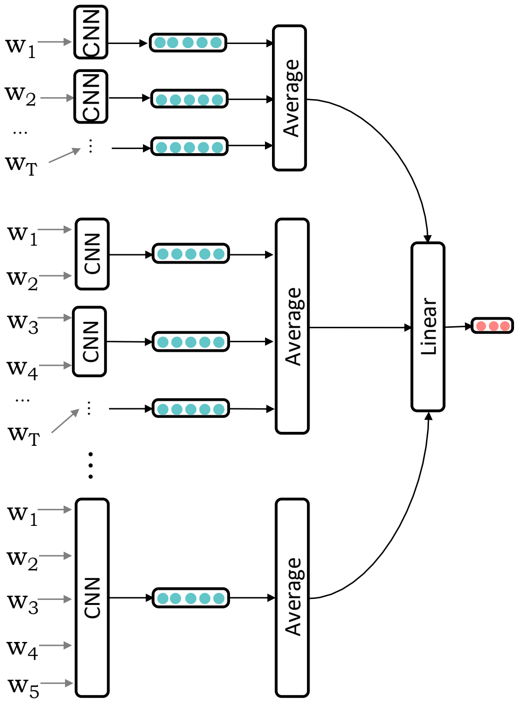


Fig. 1: Convolution and Polling operations applied to a sentence.

With the CNN model, we did four minor alterations. The first model is identical to the one showed in Figure1. In the next three, we applied the MLP model. First, passed the embeddings through the MLP model and they were the input to the Convolution Layer. Second, we applied the MLP to the output of the Convolution Layer, and after applying the Average Pooling. In third place, we applied the MLP model in the end, after the linear operation.

Long Short-Term Memory

Even though CNN has fast performance, LSTM is more successful when working with natural language processing(46), such as sequences of words expressed as time series.

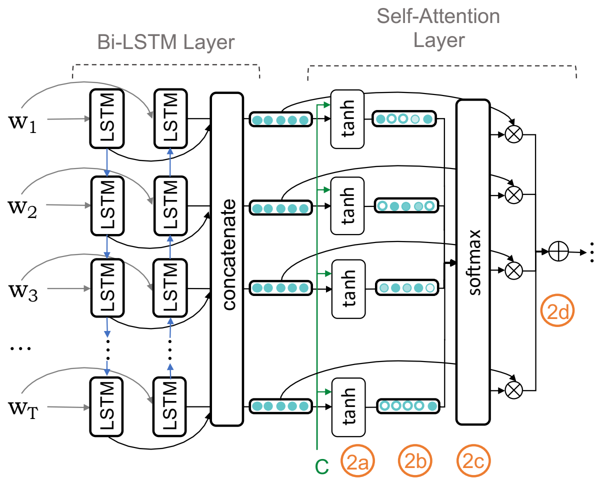


Fig. 2: BiLSTM and Attention

LSTM is a variance of a Recurrent Neural Network that uses a gated architecture and solves the vanishing gradient problem(15). LSTM uses a state vector, that allows keeping track of the state, split into two parts: the part responsible for the working memory and the "memory cell", where the essential elements of the sequence are stored. Besides, in the case of LSTM, we have to consider three types of gate. In Equation 1, we can observe all the operations that an LSTM cell require.

First, we have forget gate f, responsible for determining what information should be kept. Second, input gate i combines the previously hidden state and the current input and selects values that should be updated, through a sigmoid function. Cell gates c_t are the next stage; they do a pointwise addition that returns a new state cell with the new values that the network will compute. Ultimately, the output gate o decides what should be carried to the next hidden state. This step combines the new state and the memory cell.

$$
\begin{aligned}
s_t = R_{\text{ LSTM }}\left(s_{t-1}, x_t\right) = \left[c_t ; h_t\right] & \\
c_t &= f \odot c_{t-1} + i \odot z \\
h_t &= o \odot \tanh\left(c_t\right) \\
i &= \sigma\left(x_t W^{xi} + h_{t-1} W^{hi}\right)] \\
f &= \sigma\left(x_t W^{xf} + h_{t-1} W^{hf}\right) \\
o &= \sigma\left(x_t W^{xo} + h_{t-1} W^{ho}\right) g \\
&= \tanh\left(x_t W^{xz} + h_{t-1} W^{hz}\right) \\
y_t = O_{\text{LSTM}}\left(s_t\right) = h_t &
\end{aligned}
\tag{1}
$$

To enhance the position of each word in the sentence (38), we choose to use a Bidirectional LSTM (BiLSTM). The idea is to have two LSTMs travelling through the sentence at the same time, one that encodes the sentence left to right and, separately, other that travels from the end to the beginning of the sentence. In the end, we concatenate these two representations. This is translated into the BiLSTM Layer of the Figure2. However, as Yin et. al(46) referred in their paper, tracing the hole sentence with an LSTM can disregard the keywords. So, align with LSTM, we also used a Self-Attention Layer.

Attention

Attention was introduced by Bahdanau et al.(3) to solve translation. They proposed the use of a layer that gives attention to each source sentence and determines what parts are more relevant to achieve the expected output, even when the sentences show to be reasonably long. In other words, the decoder receives an additional weighted input that determines which tokens are necessary to pay more attention, in each time step.

Later, Vaswani et al. (42) showed that self-attention mechanisms are not only companions of other well-known machine learning models. They proposed the Transformer, a learning-based translation mechanism based on a Multi-Head Self-Attention. The model outperformed previous models with faster training time.

In our models, we used a Keras SeqSelfAttention layer with a sigmoid attention activation. This layer can be translated into the Self-Attention Layer from Figure 3a and the Equation 2.

$$
h_i, j = \tanh\left(x_i^{\top} W_1 + x_j^{\top} W_x + b_i\right) \tag{2a}
$$

$$
e_{i,j} = \sigma\left(W_a h_{i,j} + ba\right) \tag{2b}
$$

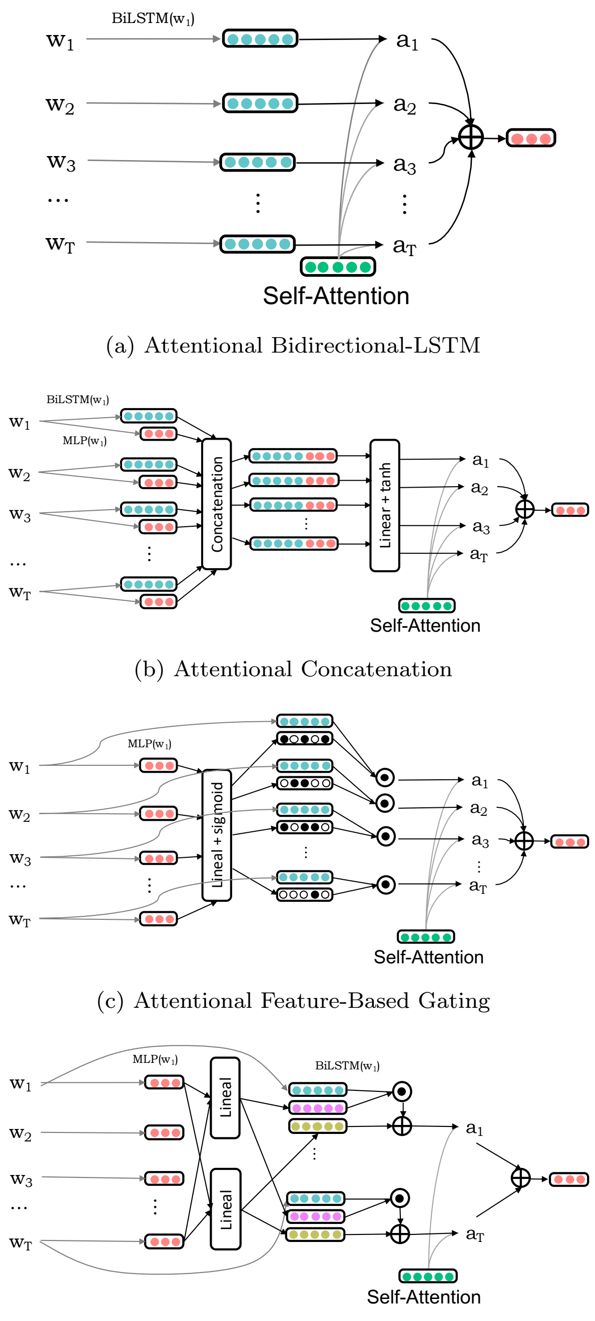
$$
a_i = \operatorname{softmax}\left(e_i\right) \tag{2c}
$$

$$
\mathrm{self}_{\mathrm{a}}\mathrm{ttention}_i = \sum_j a_{i,j} x_j \tag{2d}
$$

In Self-Attention, it is first necessary to calculate $h_{i,j}$ (2a) by summing the values of the current position and the previous, all previously multiplied by a weight matrix. After, multiplying the values by the alignment weights, we get the alignment scores (2b). On 2c, we apply softmax to the attention scores, for the values to vary between 0 and 1 and determine the probability of each given the word. At the end (2d), a_i, the amount of attention j^th should pay to i^{th} input, and summing all the results.

LSTM Models

First, we considered models with LSTM layer and a Self-Attention Layer, as we can visualize in Figure 3a. We also analysed an alteration to this model, instead of



(a) Attentional Bidirectional-LSTM

(b) Attentional Concatenation

(c) Attentional Feature-Based Gating

(d) Attentional Affine Transformation

Fig. 3: Proposed models applying Self-Attention and BiLSTM Layers.

receiving the embeddings as the input, we applied the pre-trained MLP to all the embeddings and provided the results of the operation to the LSTM Layer.

We also produced three models inspired on the work developed by Margatina et all.(28). This models were given the names they had in this paper.

Attentional Concatenation

The Attentional Concatenation model, Figure 3b and Equation 3, we calculate the BiLSTM of each embedding. In parallel, with applied the MLP pre-trained model for every word of the sentence. Then, we proceed to the concatenation of both operations and pass that concatenation through a Self-Attention Layer. In the end, calculate a Dense Layer, with three dimensions, to predict the three emotional dimensions.

$$x_1 = \tanh\left(W_c\left[\text{BiLSTM}\left(w_1\right) \| MLP\left(w_i\right)\right] + b_c\right)) \tag{3a}$$

operations 2a - 2d

$$d = l \cdot 3 + b \tag{3b}$$

Attentional Feature-Based Gating

The second method, described in Figure 3c and Equation 4, we apply the MLP pre-trained model to the word embeddings and later use a linear plus sigmoid operations. Appling the gating mechanism, by applying the sigmoid function, we will have a mask-vector where each value varies between 0 and 1 that will later be applied to the embeddings of each word by an element-wise multiplication, ⊙. Lastly, we used a Self-Attention Layer.

$$f_g\left(h_i, \text{MLP}\left(w_i\right)\right) = \sigma\left(W_g \, \text{MLP}\left(w_i\right) + b_g\right) \odot h \tag{4}$$

Attentional Affine Transformation

In the final model 3d, the feature-wise affine transformation is applied; in other words, a normalization layer preserving collinearity and ratios of distances. Primarily, we apply the pretrained MLP model to the word embeddings, and enforce a scaling and shifting vector to the results of the MLP. This model, initially inspired by Perez et al.(32), allow to capture dependencies between features by a simple multiplicative operation. The results of the linear operation γ over the MLP results are later multiplied element-wise with the results from the BiLSTM Layer over the embeddings. After, we add these values to β, and apply a Self-Attention Layer.

$$f_a\left(h_1, \text{MLP}\left(w_i\right)\right) = \gamma\left(\text{MLP}\left(w_i\right)\right) \odot h_i + \beta\left(\text{MLP}\left(w_i\right)\right) \tag{5a}$$

$$\gamma(x) = W_\gamma x + b_\gamma \tag{5b}$$

$$\beta(x) = W_\beta x + b_\beta \tag{5c}$$

4 Experimental Evaluation

In this section are described the experiments conducted to infer sentiment from textual utterances. The section is divided into Models Exploring Statistics and Models Exploring Machine Learning.

In the third set of experiments, it was necessary to access the result of a statistical models to predict the sentiment of textual utterance. One of the models that had a better performance was the MLP. Hence, an MLP was pre-trained with seven datasets in different languages. In Figure 4 it is established a correlation between the dimensional distribution of the datasets.

Three English datasets were used. The Affective Norms for English Words (Anew) (5), composed of 1,034 unique words. The early work on sentiment analysis is annotated considering the three dimensions of valence, arousal and dominance considering values between 1 and 9. (44) (Warriner) extended the previous dataset, collecting 13,915 English lemmas, also including the three dimensions. For a richer dataset, data such as gender and education level was recorded, among others. (39) (Glasgow) provided a dataset with 5,553 English words with nine dimensions identified for each, including the three dimensions valence, arousal and dominance. This dataset presents a worse spatial distribution, considering the two previous English datasets, of the words through the dimensions.

Also two Spanish datasets were considered. Redondo et al. (35) (Es Redondo) translated 1,034 Spanish words from the ANEW dataset and provided rating based on 720 annotators, also rated into the three dimensions. Through the thesis, we will call this dataset Redondo dataset. (41) (Es ANEW) expanded the amount of Spanish emotional datasets by rating 14,031 words. However, since the authors considered there was a strong correlation between valence and dominance, they chose to evaluate the words considering only valence and arousal. Through the thesis, the dataset will be called Spanish ANEW.

It was also important to consider other languages. (37) (De ANEW) created an adaptation of the ANEW dataset, ANGST. A total of 1,003 words were rated considering six dimensions (i.e. valence, arousal, dominance, arousal rated with a different metric, imageability and potency). (31) (It ANEW) also translated all the words of the original English ANEW dataset, this time into Italian, and added some more making a total of 1,121 Italian words. The annotators rated the words through the three dimensions but also added psycholinguistic indexes. (40) (Pt ANEW) provided an adaptation of the ANEW dataset. A total of 958 college students evaluated the transçated word considering the three dimensions. (18) (Pl ANEW) also translated and extended the ANEW dataset to Polish. Apart from the three dimensions, they also added a few parameters (i.e. importance, origin, concreteness).

Since one of the datasets do not have the dominance dimension, it was necessary to add a new column on the Spanish Redondo dataset ((35)), for dominance, filled with -1 and create a custom loss function. Whenever the dominance dimension is -1, the function will return zero, preventing the model to "learn" form those values.

4.1 Models Exploring Statistics

In this experiment, the goal was to observe what were the models that had better performance and compare them to more complex models. All the models that were tested in these experiments are described in section above.

Moreover, the results obtained are described on Table 1.Despite the simplicity of the model Average (i.e. the MLP model is applied to each word of the text and an average of all the outputs is calculated to deliver a final output), it was the model that showed a better performance of the word-level solutions in almost every dataset.

4.2 Models Exploring Machine Learning

Now, it was necessary to compare the results obtained with the set of experiments considering text-level sentiment prediction. To validate the models, it was necessary to conduct experiments using cross-validation.

Cross-validation is a method of allowing to validate a model (e.g. by calculating its precision), dividing a dataset into splits, usually between 2 and 5. A number of those splits are used to train the model, and the other is used to validate it. Considering that in these experiments, we are considering several datasets, it was necessary to divide each dataset equally between the splits. Using cross-validation with multiple datasets can be translated into Figure 5.

For the experiment and considering the amount of time required to train each model (i.e. considering 200 epochs), I choose to divide the datasets between two splits. In the end, each split had the same amount of each dataset. Table 2 displays the results for each model through each dataset, considering Pearson's correlation, MAE and MSE.

Through Table 2, it is possible to conclude that comparing the LSTM and CNN simple models, the LSTM shows a better performance in every dataset. Based on the explanations of section above, CNN performs better with classification tasks and LSTM with regression tasks. It is also possible to observe the variance between the values of the LSTM with and without the MLP layer of weights. It was expected that a pre-trained MLP layer would help to provide better predictions. However, by comparing the Pearson correlation of both

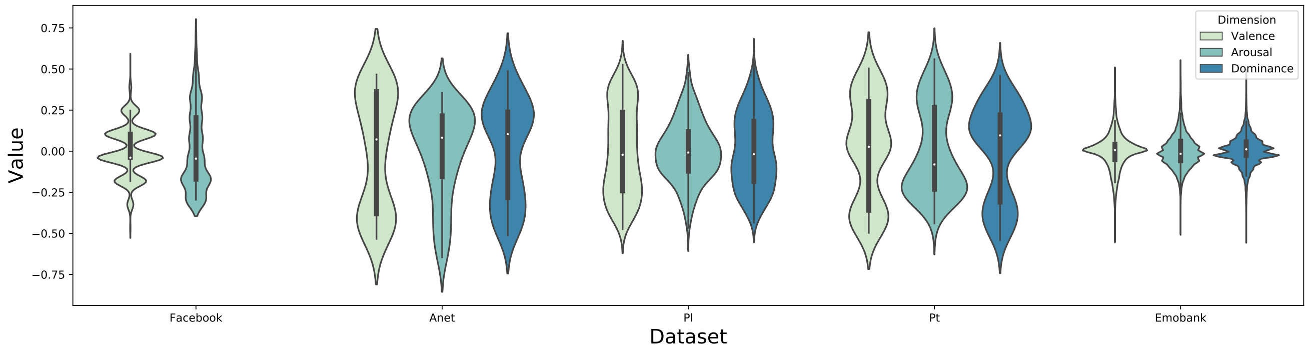


Fig. 4: Comparison of the dimensional distribution of the datasets in several languages.

		Pt		Pl		Emobank		ANET		Fb	
		Pearson	MAE	Pearson	MAE	Pearson	MAE	Pearson	MAE	Pearson	MAE
MLP Average	V	**0.686**	0.234	**0.499**	0.227	**0.359**	0.086	0.639	**0.301**	**0.384**	0.154
	A	0.511	0.216	0.222	**0.160**	**0.152**	0.101	**0.542**	0.319	**0.111**	0.234
	D	0.470	**0.238**	0.312	0.187	0.058	0.093	0.261	0.263	-	-
Average MLP	V	0.625	**0.232**	0.429	0.226	0.284	0.073	**0.697**	0.312	0.298	**0.132**
	A	0.342	0.218	0.109	0.187	0.122	0.089	0.433	0.355	0.790	0.237
	D	**0.579**	0.234	**0.436**	0.194	**0.123**	0.122	**0.622**	**0.258**	-	-
Pooling Average MLP	V	0.482	0.256	0.453	0.231	0.201	0.091	0.491	0.323	0.192	0.149
	A	0.187	0.231	0.166	**0.160**	0.110	0.122	0.420	**0.316**	0.79	0.250
	D	0.310	0.257	0.358	**0.183**	0.057	**0.092**	0.397	0.277	-	-
Pooling MLP Average	V	0.537	0.249	0.456	0.231	0.224	0.094	0.492	0.323	0.193	0.148
	A	0.266	0.230	0.168	**0.160**	0.098	0.130	0.420	**0.316**	0.82	0.244
	D	0.405	0.263	0.359	**0.183**	0.068	0.100	0.396	0.360	-	-
MLP Pooling Avg	V	0.339	0.317	0.402	**0.222**	0.083	**0.071**	0.605	0.312	0.137	0.161
	A	0.330	0.253	**0.335**	0.188	0.029	**0.088**	0.515	0.336	0.152	**0.208**
	D	0.219	0.342	0.256	**0.183**	0.039	0.182	0.327	0.268	-	-

Table 1: Results obtained for statistical sentiment prediction of textual utterances, in terms of Pearson's correlation coefficient and MAE.

experiments, it is possible to observe worse results when using the MLP layer.

The results obtained through a cross-lingual experiment were are good. However, it is necessary to consider that the model would have a better performance if it was trained only for and with training data from only one language. The dimension that was more difficult to tackle was arousal, especially in the Facebook dataset.

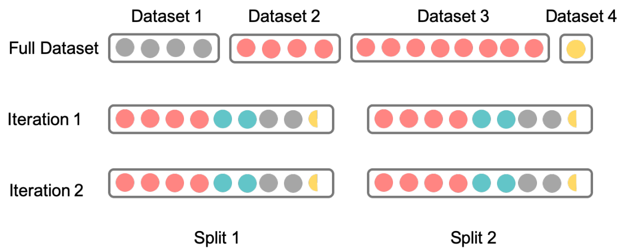


Fig. 5: Example of cross-validation with multiple datasets.

All the three last models had a great performance, compared to the rest of the models. Even though in some datasets (i.e. such as the Portuguese) the results were similar to the Average model (i.e. in Table 1), it is possible to see a great improvement on bigger datasets, such as the Facebook.

The work of (23), with a BiLSTM model, it is possible to observe an MSE correlation on the Facebook dataset of 0.990 and 3.550 for valence and arousal respectively. Comparing to the results obtained with the Attention Feacture Based model for the Facebook dataset, it is possible to conclude that this model was able to outperform their results.

Considering the results obtained by (1) (i.e. with a Pearson's correlation of 0.727 and 0.355 for the Facebook dataset, and 0.635 and 0.375 for the Emobank dataset, for valence and arousal respectively), it is possible to observe that the Attention Concat model performed comparably. This work even showing better values for the dimension arousal than the work from (1). Ultimately, it is possible to conclude that the Attention

		Pt			Pl			Emobank			ANET			Fb		
		Pearson	MAE	MSE	Pearson	MAE	MSE	Pearson	MAE	MSE	Pearson	MAE	MSE	Pearson	MAE	MSE
LSTM	V	0.641	0.184	0.059	**0.507**	**0.184**	0.055	**0.536**	0.070	0.009	**0.769**	0.207	0.059	0.547	0.100	0.018
	A	0.608	**0.164**	0.047	0.333	0.166	0.034	**0.333**	0.088	0.013	**0.617**	0.188	0.053	0.494	0.177	0.060
	D	0.576	**0.164**	0.056	0.445	0.149	0.042	0.092	0.120	0.065	0.439	0.231	0.082	-	-	-
MLP + LSTM	V	0.319	0.246	0.087	0.258	0.225	0.073	0.150	0.276	0.012	0.236	0.316	0.120	0.065	0.126	0.026
	A	0.232	0.241	0.071	0.108	0.146	0.034	0.016	0.297	0.013	0.254	0.282	0.097	0.126	0.235	0.081
	D	0.345	0.232	0.069	0.296	0.192	0.054	0.022	0.552	0.093	0.112	0.375	0.252	-	-	-
CNN	V	0.632	0.228	0.062	0.415	0.211	0.072	0.434	0.070	0.021	0.672	0.261	0.092	0.495	0.102	0.020
	A	0.312	0.241	0.050	0.241	0.148	0.059	0.170	.0.087	0.032	0.493	0.221	0.168	0.260	0.212	0.058
	D	0.427	0.234	0.063	0.247	0.235	0.109	0.040	0.258	0.075	0.261	0.329	0.091	-	-	-
MLP + CNN	V	0.584	0.236	0.076	0.397	0.212	0.067	0.466	**0.069**	0.009	0.657	0.249	0.087	0.501	0.109	0.019
	A	0.345	0.221	0.063	0.281	0.146	0.034	0.136	0.089	0.013	0.536	0.204	0.060	0.316	0.215	0.065
	D	0.419	0.227	0.078	0.282	0.202	0.067	0.040	0.251	0.085	0.167	0.410	0.302	-	-	-
CNN + MLP	V	0.552	0.223	0.066	0.395	0.215	0.066	0.449	0.071	0.009	0.523	**0.080**	0.066	0.485	0.107	0.019
	A	0.343	0.219	0.034	0.197	0.145	0.034	0.214	0.088	0.013	0.393	**0.114**	0.058	0.315	0.215	0.067
	D	0.342	0.227	0.061	0.243	**0.147**	0.061	0.066	0.182	0.076	0.408	0.291	0.149	-	-	-
Attention Concat	V	**0.691**	**0.177**	0.057	0.435	0.202	0.064	0.507	0.073	0.010	0.649	0.238	0.007	**0.561**	0.101	0.019
	A	**0.620**	0.165	0.046	0.297	0.144	0.035	0.302	0.089	0.014	0.481	0.209	0.051	**0.565**	0.176	0.052
	D	**0.663**	0.167	0.049	0.348	0.182	0.050	**0.363**	0.122	0.074	0.283	0.276	0.004	-	-	-
Attention	V	0.641	0.184	0.050	0.501	0.192	0.059	0.531	**0.069**	0.001	0.680	0.226	0.056	0.557	**0.098**	0.021
Feacture	A	0.608	0.164	0.042	**0.391**	**0.137**	0.031	0.320	**0.083**	0.014	0.538	0.198	0.051	0.545	**0.174**	0.057
Based	D	0.576	0.173	0.058	**0.470**	0.160	0.043	0.082	**0.116**	0.065	0.479	**0.217**	0.084	-	-	-
Attention	V	0.569	0.206	0.072	0.434	0.206	0.065	0.501	0.074	0.010	0.728	0.225	0.070	0.523	0.108	0.057
Affine	A	0.540	0.177	0.050	0.268	0.148	0.036	0.270	0.092	0.015	0.608	0.189	0.056	0.491	0.184	0.436
Transformation	D	0.473	0.218	0.075	0.338	0.180	0.051	0.075	0.143	0.067	**0.481**	0.266	0.119	-	-	-

Table 2: The prediction of valence, arousal and dominance with several models. The training and testing data are textual utterances form datasets English, Polish and Portuguese.

Concat model has a great performance, even compared to models that were trained for one language.

The results obtained by a stat-of-the-art work conducted by (13), with a Pearson correlation of 0.553 and 0.348 for the Emobank dataset and 0.725 and 0.925 for Facebook (i.e. for the dimensions valence and arousal respectively). The results were obtained using a model composed by Bi-LSTM+MP+Attention. Similar to my results using the Attention Feacture Based model, that obtained results of 0.531 and 0.320 for Emobank, 0.557 and 0.545 for Facebook. Comparing the results and considering that my model performed a little lower, but being trained with several idioms, the lower performance can be justified. It is possible to also see an improvement in my model regarding the MAE and MSE values, where the Emobank obtained 0.069 and 0.003 (i.e. MAE and MSE respectively for the valence dimension), 0.083 and 0.013 (i.e. MAE and MSE respectively for the arousal dimension); where (13) obtained 0.268 and 0.127 (i.e. MAE and MSE respectively for the valence dimension), 0.251 and 0.104 (i.e. MAE and MSE respectively for the arousal dimension).

Also, comparing the LSTM (with a mean of the three dimensions of 0.61 for ANET, 0.43 for Pl and 0.61 for Pt) comparing to the (8) (0.73 for the ANET, 0.56 and 0.65 for Pt), shows that a the LSTM models of this thesis performes slightly lower. However, the model proposed by (8) is targeted for each language separately, considering this, it is possible to conclude that, even with a slightly inferior results, the LSTM model presented in this thesis, shows promising results.

In these set of experiments, I reveal the results obtained with the models described previously, where the trained models were trained with cross-validation of two splits using datasets in several idioms. The Polish dataset showed worth results compared to the rest of the datasets. The models presented in this thesis performed better than most state-of-the-art works, even considering that these models are not trained to tackle only one language. However, it would be interesting to compare this work considering datasets in more languages.

5 Conclusions and Future Work

This research aimed to understand if it was possible to a machine learning model to quantify emotion regarding valence, arousal and dominance in multiple languages. To understand if an ML model can quantify sentiment in textual utterances of multiple languages, it was necessary to set the following secondary questions. What method provides better results: a word or text-level sentiment prediction for text? Are CNN's or LSTM's better for sentiment prediction? Do models with pre-trained MLP perform better or worst? What are the models that perform better in this scenario?

To answer the secondary questions, several models where created. An MLP was pre-trained with lexicons from six different languages. To infer the need to access all the syntactic structure to infer sentiment form a text, four models that do not take into consideration the syntactic structure and do not require training were created. Furthermore, eight trainable models were conceived (i.e. LSTM, MLP+LSTM, CNN, MLP+CNN, CNN+MLP, Attention Concat, Attention Feacture Bassed, Attention Affine Transformation), validated with two-fold cross-validation.

The results show that three trained models performed better (Attention Concat, Attention Feacture Bassed, Attention Affine Transformation); however, the Average word-level prediction model also showed promising results. LSTM's tend to perform slightly better than CNN models. The difference was more evident in the arousal dimension. However, when the CNN and LSTM model were aligned with the pre-trained MLP, the results decreased, showing that a pre-trained MLP can decrease the performance of the model.

Overall, this work shows promising results when inferring sentiment, even in several languages. The main contribution of this work relies, first on the significant amount of models that were validated to infer how to extract sentiment from both words and textual utterances. There are few works on sentiment quantification, in particular, considering the dimensional way of quantifying sentiment. This thesis provides three trained models and one word-level model that show promising results compared to the state-of-the-art.

Although our experiments have shown promising results with the usage of unsupervised multilingual word embeddings (umwe) for leveraging English data with the purpose of estimating lexical norms for other languages, umwe is a framework that only works with MUSE experiment that faces some instability issues. As it was already stated MUSE has some limitations with some languages.

This study provides, as theoretical implications, a comparison between statistical models and machine learning models. As well as, a comprehensive comparison between these machine learning models. Possible practical applications to the findings in this study could be to monitor informal political online discussions and to lead to a better understanding of hate speech on social media. As well as, to better understand the mass opinion on trendy subjects.

For future work, it could be interesting to extend the experiments reported here, considering also other languages and other types of lexical norms (e.g., leveraging data from the Bristol norms for age of acquisition, imageability, and familiarity), other types of forecasting models (e.g., different types of ensemble approaches, combining different modelling alternatives and choosing the best combination through cross-validation). As well as the combination of skip-ngram word embeddings with other types of features, such as the incorporation of features based on word frequency, word length or orthographic similarity. Besides fasttext embeddings, there are other distributional word representations that could also have been used in these thesis tests for comparison. Recent studies suggest that, after careful hyper-parameter tuning, there are no global advantages in any of the proposals from the recent literature. Still, for future work, it could be also interesting to experiment with word embeddings trained on different types of corpora (e.g., on social media data, that is perhaps more reflective of people's attitudes and emotions) and/or relying on different approaches, such as the GloVe method.

References

1. Akhtar, S., Ghosal, D., Ekbal, A., Bhattacharyya, P., Kurohashi, S.: All-in-one: Emotion, sentiment and intensity prediction using a multi-task ensemble framework. IEEE Transactions on Affective Computing (2019)
2. Alswaidan, N., Menai, M.E.B.: KSU at SemEval-2019 Task 3: Hybrid Features for Emotion Recognition in Textual Conversation. In: Proceedings of the 13th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2019, Minneapolis, MN, USA, June 6-7, 2019, pp. 247–250 (2019)
3. Bahdanau, D., Cho, K., Bengio, Y.: Neural Machine Translation by Jointly Learning to Align and Translate. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)
4. Binali, H., Wu, C., Potdar, V.: Computational approaches for emotion detection in text. In: 4th IEEE International Conference on Digital Ecosystems and Technologies, pp. 172–177. IEEE (2010)
5. Bradley, M.M., Lang, P.J.: Affective norms for english words (anew): Instruction manual and affective ratings. Tech. rep., Technical report C-1, the center for research in psychophysiology ... (1999)
6. Buechel, S., Hahn, U.: Emotion Analysis as a Regression Problem-Dimensional Models and Their Implications on Emotion Representation and Metrical Evaluation. In: ECAI, pp. 1114–1122 (2016)
7. Buechel, S., Hahn, U.: Word Emotion Induction for Multiple Languages as a Deep Multi-Task Learning Problem. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers), pp. 1907–1918 (2018)
8. Buechel, S., Sedoc, J., Schwartz, H.A., Ungar, L.: Learning neural emotion analysis from 100

observations: The surprising effectiveness of pre-trained word representations. arXiv preprint arXiv:1810.10949 (2018)

9. Calvo, R.A., Mac Kim, S.: Emotions in text: dimensional and categorical models. Computational Intelligence **29**(3), 527–543 (2013)
10. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. Journal of Machine Learning Research **12**, 2493–2537 (2011)
11. Ekman, P.: An argument for basic emotions. Cognition & Emotion **6**(3-4), 169–200 (1992)
12. Ekman, P., Friesen, W.V.: Constants across cultures in the face and emotion. Journal of Personality and Social Psychology **17**(2), 124 (1971)
13. Godinho, J.D.F.: Extraction, attribution, and classification of quotations in newspaper articles. Ph.D. thesis, University of Lisbon (2018)
14. Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T.: Learning word vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
16. Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology **160**(1), 106–154 (1962)
17. Hutto, C.J., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Eighth international AAAI conference on weblogs and social media (2014)
18. Imbir, K.K.: Affective norms for 4900 polish words reload (anpw_r): assessments for valence, arousal, dominance, origin, significance, concreteness, imageability and, age of acquisition. Frontiers in psychology **7**, 1081 (2016)
19. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. CoRR **abs/1607.01759** (2016). URL http://arxiv.org/abs/1607.01759
20. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188 (2014)
21. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
23. Kratzwald, B., Ilic, S., Kraus, M., Feuerriegel, S., Prendinger, H.: Decision support with text-based emotion recognition: Deep learning for affective computing. arXiv preprint arXiv:1803.06397 (2018)
24. LaBrie, R.C., Louis, R.D.S.: Information Retrieval from Knowledge Management Systems: Using Knowledge Hierarchies to Overcome Keyword Limitations. In: 9th Americas Conference on Information Systems, AMCIS 2003, Tampa, FL, USA, August 4-6, 2003, p. 333 (2003)
25. LeCun, Y., Bengio, Y., et al.: Convolutional networks for images, speech, and time series. The Handbook of Brain Theory and Neural Networks **3361**(10), 1995 (1995)
26. Ma, C., Prendinger, H., Ishizuka, M.: Emotion Estimation and Reasoning Based on Affective Textual Interaction. In: Affective Computing and Intelligent Interaction, First International Conference, ACII 2005, Beijing, China, October 22-24, 2005, Proceedings, pp. 622–628 (2005). DOI 10.1007/11573548_80
27. Malheiro, R., Oliveira, H.G., Gomes, P., Paiva, R.P.: Keyword-based Approach for Lyrics Emotion Variation Detection. In: Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 1: KDIR, Porto - Portugal, November 9 - 11, 2016., pp. 33–44 (2016). DOI 10.5220/0006037300330044
28. Margatina, K., Baziotis, C., Potamianos, A.: Attention-based Conditioning Methods for External Knowledge Integration. arXiv preprint arXiv:1906.03674 (2019)
29. Mikolov, T., Deoras, A., Povey, D., Burget, L., Cernocký, J.: Strategies for training large scale neural network language models. In: 2011 IEEE Workshop on Automatic Speech Recognition & Understanding, ASRU 2011, Waikoloa, HI, USA, December 11-15, 2011, pp. 196–201 (2011). DOI 10.1109/ASRU.2011.6163930
30. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed Representations of Words and Phrases and their Compositionality. CoRR **abs/1310.4546** (2013). URL http://arxiv.org/abs/1310.4546
31. Montefinese, M., Ambrosini, E., Fairfield, B., Mammarella, N.: The adaptation of the affective norms for english words (anew) for italian. Behavior research methods **46**(3), 887–903 (2014)
32. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: Visual reasoning with a general conditioning layer. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)

33. Perugini, M., Bagozzi, R.P.: The role of desires and anticipated emotions in goal-directed behaviours: Broadening and deepening the theory of planned behaviour. British Journal of Social Psychology **40**(1), 79–98 (2001)
34. Preoţiuc-Pietro, D., Schwartz, H.A., Park, G., Eichstaedt, J., Kern, M., Ungar, L., Shulman, E.: Modelling valence and arousal in facebook posts. In: Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 9–15 (2016)
35. Redondo, J., Fraga, I., Padrón, I., Comesaña, M.: The spanish adaptation of anew (affective norms for english words). Behavior research methods **39**(3), 600–605 (2007)
36. Ruck, D.W., Rogers, S.K., Kabrisky, M., Oxley, M.E., Suter, B.W.: The multilayer perceptron as an approximation to a bayes optimal discriminant function. IEEE Transactions on Neural Networks **1**(4), 296–298 (1990)
37. Schmidtke, D.S., Schröder, T., Jacobs, A.M., Conrad, M.: Angst: Affective norms for german sentiment terms, derived from the affective norms for english words. Behavior research methods **46**(4), 1108–1118 (2014)
38. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing **45**(11), 2673–2681 (1997)
39. Scott, G.G., Keitel, A., Becirspahic, M., Yao, B., Sereno, S.C.: The glasgow norms: Ratings of 5,500 words on nine scales. Behavior research methods **51**(3), 1258–1270 (2019)
40. Soares, A.P., Comesaña, M., Pinheiro, A.P., Simões, A., Frade, C.S.: The adaptation of the affective norms for english words (anew) for european portuguese. Behavior research methods **44**(1), 256–269 (2012)
41. Stadthagen-Gonzalez, H., Imbault, C., Sánchez, M.A.P., Brysbaert, M.: Norms of valence and arousal for 14,031 spanish words. Behavior research methods **49**(1), 111–123 (2017)
42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is All you Need. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp. 5998–6008 (2017)
43. Wang, J., Yu, L.C., Lai, K.R., Zhang, X.: Community-based weighted graph model for valence-arousal prediction of affective words. IEEE/ACM Transactions on Audio, Speech, and Language Processing **24**(11), 1957–1968 (2016)
44. Warriner, A.B., Kuperman, V., Brysbaert, M.: Norms of valence, arousal, and dominance for 13,915 english lemmas. Behavior research methods **45**(4), 1191–1207 (2013)
45. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics (2005)
46. Yin, W., Kann, K., Yu, M., Schütze, H.: Comparative study of cnn and rnn for natural language processing. arXiv preprint arXiv:1702.01923 (2017)
47. Zahiri, S.M., Choi, J.D.: Emotion Detection on TV Show Transcripts with Sequence-based Convolutional Neural Networks. arXiv preprint arXiv:1708.04299 (2017)

Sofia Aparicio is an M.Sc. student in the Computer Science and Engineering Department of IST-UL, and a junior researcher at the Decision Support Systems (IDSS) laboratory of INESC-ID. Previously, she got her undergraduate degree on Computer Science and Engineering, also at IST-UL. Her M.Sc. thesis research relates to the study of emotional properties associated to textual utterances.

Bruno Martins got his Ph.D. in Computer Science in the year of 2009, from the University of Lisbon (UL). He is currently an Assistant Professor at the Computer Science and Engineering Department of IST-UL, where he teaches courses related to Databases and Data Management, Information Retrieval, and Information Extraction. He is also a researcher at the Information and Decision Support Systems (IDSS) laboratory of INESC-ID, where he mostly works on problems related to the general areas of information retrieval, text mining, and the geographical information sciences.