Computational Linguistics in the Netherlands Journal 8 (2018) 99–121 Submitted 06/2018; Published 12/2018

Semi-Supervised Emotion Lexicon Expansion with Label Propagation

Mario Giulianelli* MARIO.GIULIANELLI@STUDENT.UVA.NL
Daniël de Kok** DANIEL.DE-KOK@UNI-TUEBINGEN.DE

* *Institute for Logic, Language and Computation, University of Amsterdam*

** *Department of General and Computational Linguistics, University of Tübingen*

Abstract

The task of emotion classification has traditionally been addressed using two different but complementary approaches: lexicon-based approaches typically have a wider coverage of emotion-bearing words whereas corpus-based approaches learn to use contextual cues. It should not come as a surprise that these methods have been used jointly to exploit the strengths of both.

However, a combination of the two techniques still suffers from the relatively limited size of the available linguistic resources. In this work, we introduce a novel variant of the Label Propagation algorithm (Zhu and Ghahramani 2002) to extend the coverage of an existing emotion lexicon. In order to do so, we construct a fully connected graph wherein words are vertices and the edges are weighted by the geometric proximity of distributional word representations. The vertices that correspond to words that occur in an emotion lexicon are initialised using the emotion distribution indicated in the lexicon. Then, the label propagation algorithm is used to derive emotion distributions for words that do not occur in the lexicon. Finally, we propose batched label propagation: an optimisation procedure which makes expansion tractable for large vocabularies.

In our experiments, we compare four emotion classifiers: the model of Mohammad and Kiritchenko (2015); a bidirectional LSTM model; a bidirectional LSTM model using an emotion lexicon; a bidirectional LSTM model using the extended emotion lexicon derived through label propagation. Our results show that the classifier that uses the expanded emotion lexicon outperforms the other models on the two deployed emotion classification benchmarks.

1. Introduction

On online platforms, users express their opinions and share their experiences, thus they generate a complex network of mutual influence. Reviews of goods and services, political views and commentaries, as well as recommendations of job applicants exemplify how Web content can impact the decision-making process of consumers, voters, companies, and other organisations (Pang et al. 2008). Users' opinions have been classified with regard to, among other aspects, the polarity of the sentiment they express and to the basic emotions they evoke.

In the field of polarity annotation there exist two main approaches to automatically extract sentiment: corpus-based and lexicon-based. The same dichotomy exists for the detection of affectual orientation in text. The corpus-based approach frames emotion analysis as a supervised classification task and requires emotion-annotated corpora. Supervised emotion classifiers are typically context-dependent, so they tend to perform well in the domain of the training corpus but they can be mediocre when tested on other domains. The lexicon-based approach considers the orientation of single words and phrases in the document. As well as the inherent context independence (Taboada et al. 2011), which negatively affects precision, another important limitation of this approach is the small size of the available lexical resources. We propose a new reproducible method for emotion lexicon expansion that directly addresses the second shortcoming and can be leveraged to improve the overall performance of an emotion classifier.

In the proposed framework, emotion-specific word embeddings are learned from a corpus using an LSTM neural network. Each text in the corpus is labelled with six basic emotions (anger, disgust, fear, joy, sadness, and surprise). Then, the derived vector space model is used to expand an existing emotion lexicon via a novel variant of the semi-supervised Label Propagation algorithm (Zhu and Ghahramani 2002) which is tailored to distributed word representations. Finally, batch gradient descent is proposed as a way to accelerate the optimisation of label propagation and to make it feasible for large graphs (i.e. for large vocabularies). Our method requires two types of resources: an emotion-labelled corpus and an emotion lexicon. We use, respectively, the Hashtag Emotion Corpus (Mohammad and Kiritchenko 2015), a collection of tweets labelled with Ekman's six basic emotions (Ekman 1992), and the NRC Emotion Lexicon (Mohammad and Turney 2013).

The rest of this paper is structured as follows. We begin in Section 2 with a review of related work in the areas of emotion classification and lexicon expansion. Then, the proposed lexicon expansion method and the optimisation of specialised word embeddings are introduced (Section 3). Section 4 presents an analysis of the employed linguistic resources. The experiments performed to learn emotion-specific embeddings and to expand the lexicon are reported in Section 5 along with our intrinsic and extrinsic evaluation in an emotion classification task (Section 6). Section 7 concludes and proposes new research ideas.

The software related to this paper is open-source and available at `github.com/Procope/emo2vec`.

2. Background

2.1 Emotion classification

Sentiment analysis refers to the automatic detection of a user's affective attitude toward a topic. More commonly, the goal of the field is considered to be that of determining the *polarity* of phrases, sentences, and documents. This is often referred to as *polarity* or *valence annotation*, that is the classification of pieces of text as positive, negative, or neutral. It is common to extend the number of valence classes to at least five; this finer-grained analysis also typically includes *very positive* and *very negative* in the unidimensional polarity scale. In contrast, *emotion classification* consists of assigning pieces of text to one or more classes of emotions; its goal is to gain in interpretative and explanatory power by increasing the number of dimensions used to represent affectual orientation.

The inter-annotator agreement studies conducted by Strapparava and Mihalcea (2007) for the SemEval headlines corpus established a human-level upper bound to the accuracy of emotion classification algorithms. Table 1 shows the agreement scores in terms of Pearson correlation, as they can be found in the original paper. If trained annotators agree with a simple average r of 53 (the frequency-based average r is 43), it is plausible that untrained respondents would show an even lower level of agreement. An attempt to answer this question empirically can be found in Section 6.2, which describes the results of a survey that tests human ability to classify short paragraphs into emotions.

The remainder of this section describes the two approaches that are traditionally used to address the task of emotion classification—lexicon-based and corpus-based—and discusses possible ways of

anger	disgust	fear	joy	sadness	surprise
50	45	64	60	68	36

simple average	frequency-based average
53	43

Table 1: Class-based and average Pearson correlation as a measure of inter-annotator agreement on the SemEval-2007 Affective Text Corpus (Strapparava and Mihalcea 2007).

combining their strengths. A comprehensive list of challenges for the analysis of affective text is presented by Mohammad (2015) and it includes e.g. subjective and cross-cultural differences.

Lexicon-based methods An important milestone for emotion analysis was the SemEval-2007 Affective Text task. The motivation for the task was that there seems to be a connection between lexical semantics and the way we verbally express emotions (Strapparava and Mihalcea 2007). In particular, it was argued that the emotional orientation and strength of a text (i.e. which emotion the text conveys and how intensely) are determined potentially by all words that compose the text, though, disputably, in uneven amount. In order to obtain a wide coverage of emotion-bearing words, lexicon-based methods make use of emotion dictionaries. An example of a lexicon-based method is UPAR7, a system that participated in the SemEval-2007 Affective Text task. UPAR7 parses documents and then uses the resulting dependency graphs to reconstruct what is said about the main subjects. All words in a document have emotion scores which are based on SentiWordNet (Esuli and Sebastiani 2007) and WordNet Affect (Strapparava et al. 2004) and are enriched with lexical contrast and accentuation. A higher weight is given to the score of the main subject, while the other weights are computed considering linguistic features such as negations and modal verbs. In this way, expressions that appear to be neutral can also convey affective meaning as they might be semantically related to emotional concepts.

To understand why the latter is an important property of an emotion analysis method, consider the following tweets:

(1) I want cake. I bet we don't have any.

(2) Saddened by the terrifying events in Virginia.

Although the literal meaning of (1) appears to be rather neutral, this utterance expresses a sense of frustration, which in terms of basic emotions could be translated with the labels *anger* and / or *sadness*. Conversely, (2) explicitly expresses affectual orientation yet the NRC Emotion Lexicon does not contain *sadden*, *saddened* nor *terrify*, *terrifying*. The claim that all words potentially convey affective meaning is inspirational for our work and it provides a rationale for lexicon expansion (Section 2.2 and 3.2). In Section 4 we present the Hashtag Emotion Corpus and the NRC Emotion Lexicon and use them to further argue that available lexical resources are limited if we assume that possibly all terms in a document contribute to its affective content.

A second drawback of the lexicon-based approach is again related to the labelled dictionaries used to calculate the emotional orientation of a text based on the words and phrases that constitute it. Dictionaries are static resources and they cannot differentiate between direct affective words, i.e. words that refer directly to affective states, and indirect affective words, which only have a weak connection to emotional concepts. To correctly assign labels to indirect affective words detailed, dynamic contextual information is required. To give an example, an American professional baseball player, who was criticised for his unsatisfactory performance, publicly stated:

(3) I am going to have a monster year.

The indirect affective word *monster* is used as a positive modifier but world knowledge is required to make such an inference.

Finally, consider the following headline:

(4) Beating poverty in a small way.

Its affect is rather positive yet the headline contains the direct affective words *beating* and *poverty*, which are labelled as expressions of *anger*, *disgust*, *fear*, and *sadness* in the NRC Emotion Lexicon (Section 4). Sentence (4) is an example of how lexicon-based methods cannot correctly analyse compositionality (*beating poverty*). Negations and sarcasm are other such issues, which can only be solved using methods that do not treat sentences as bags of words.

Corpus-based methods The automatic extraction of affectual orientation can also be viewed as a supervised classification problem, which requires an emotion-annotated data set and a statistical learning algorithm. This is often referred to as the *corpus-based* approach (Pang et al. 2002). In the area of corpus-based methods, researchers have proposed different systems that, having access to contextual information, have the potential to address the issues of lexicon-based techniques, such as basic compositionality—including, crucially, negation—and cases of obvious sarcasm (Strapparava and Mihalcea 2007, Mohammad and Kiritchenko 2015).

A joint approach There exist as well combinations of the two main approaches. Strapparava and Mihalcea (2008) use lexical resources (WordNet Affect) to annotate synsets representing emotions and moods: for each emotion, a list of words is generated by the corresponding synset. On the other hand, a vector space model of the British National Corpus is obtained using Latent Semantic Analysis (LSA). Not only words, but also documents and synsets are represented as vectors; the latter two are mapped into the vector space by computing the sum over the normalised LSA vectors of all the words comprised in them. Once an emotion is represented in the LSA vector space, determining affective orientation is essentially a matter of computing a similarity measure between an input word, paragraph, or text, and the prototypical emotion vectors. This was the best performing system at SemEval-2007, suggesting that dictionaries and corpora are complementary sources of information.

Indeed, further studies (Kennedy and Inkpen 2006, Andreevskaia and Bergler 2008, Qiu et al. 2009) showed that the performance of lexicon- and corpus-based approaches is complementary in terms of precision and recall. The lexicon-based method yields higher recall at the cost of low precision. Due to its static coverage of emotion-bearing words, a lexicon-based method is likely to tag an instance with a label e whenever an emotion word related to e is found, even though the text is neutral or evocative of another emotion. In contrast, supervised learning systems tend to perform poorly in terms of recall since their vocabulary is limited to the types seen in the training data and they lack an explicit indication of which affective orientation is conveyed by specific words. They reach, however, higher precision scores as they learn to use contextual cues (Yang et al. 2015).

2.2 Lexicon expansion and semi-supervised learning

The task of expanding a lexicon can be solved using a variety of methods. *Self-training* is likely the simplest approach. It consists of (i) labelling a subset of the vocabulary based on the few lexicon words, and (ii) using the newly labelled lemmas to incrementally classify larger portions of the lexicon. The drawback of self-training is that, since dictionaries are typically not very large, the first rounds of classification perform poorly thereby propagating severe, early errors throughout iterations.

An alternative method is *transductive inference* (Vapnik and Vapnik 1998). In contrast with inductive, fully supervised algorithms, which only exploit the labelled portion of the data, transductive inference takes the structure of the entire dataset into account. This evidently suits the lexicon expansion task, where only a minority of the words are labelled with emotion tags and the underlying structure of the vocabulary in feature space can be relevant to determine the emotional orientation of an unlabelled word. In particular, transductive SVMs (TSVMs) have been successfully used for text classification but they have a crucial pitfall: the TSVM optimisation problem is combinatorial. Although Joachims (1999) proposed an algorithm that finds an approximative solution using local search, in order to keep the optimisation problem tractable the size of test sets must not exceed 10,000–15,000 instances. Section 4 explains why this is an important limitation. An additional downside is that a TSVM accepts input in the form of sparse descriptors rather than dense vectors.

A more linguistically inclined approach is to compute the *semantic orientation* of words based on the PMI between tokens and emotion words—or, on Twitter, emoticons (Zhou et al. 2014). This method has been used to expand a lexicon for context-dependent polarity annotation.

The problem of lexicon expansion can also be conceived as a supervised classification task, where the words in the dictionary are used as training data. Bravo-Marquez et al. (2016) recently de-

ployed a corpus of ten million tweets (Petrovic et al. 2010) and a multi-label classifier to expand the NRC Emotion Lexicon. The proposed classifiers, Binary Relevance, Classifier Chains and Bayesian Classifier Chains, use Skip-gram embeddings as word descriptors. They are all based on word-level features which can be extracted from the Skip-gram model, or the word-centroid model. Although the word-centroid model draws information from multiple features—word unigrams, Brown clusters, POS n-grams, and Distant Polarity— word embeddings learned via the Skip-gram model were shown to significantly outperform word-centroid features at improving classification performance. These results suggest that statistical regularities in language corpora allow distributional word representations to encode affect. Yet the disproportion between lexicon words and unseen types suggests that distributional word representations could be better leveraged if they were not deployed in isolation. Semi-supervised approaches can take into account word representations as well as their relative semantic similarity, thus they seem to more naturally fit the problem of expanding a limited lexicon to a majority of unlabelled words.

A semi-supervised alternative to multi-label classification is Label Propagation, an iterative algorithm that propagates labels from labelled to unlabelled data by finding high density areas (Zhu and Ghahramani 2002). All words, labelled and unlabelled, are defined as nodes. The edge between two nodes u, v is weighted by a function of the proximity of u and v so that words that are close in the semantic space are linked by strong edges. Furthermore, all nodes are assigned a probability distribution over labels. At each iteration, labels propagate through the graph and probability mass is redistributed following a crucial principle: labels propagate faster through strongly weighted edges. An important advantage of LP is that it can be solved without iterations by computing a solution analytically. This technique appears to be the most appropriate for lexicon expansion as it can leverage dense word embeddings, their inherent context-dependence, and geometric measures of semantic similarities between words.

Task-specific word representations In the field of sentiment analysis, Tang et al. (2014) introduced a method to learn sentiment-specific word embeddings from a large annotated corpus. The proposed approach consists of extending an existing distributional embedding algorithm, the Collobert and Weston (C&W) model (Collobert et al. 2011) and produces representations that tend to enhance the predictive performance of supervised sentiment classifiers.

The algorithm combines two concurrent objectives: modelling the syntactic context of words (with the C&W loss) and learning the polarity of words based on their affective sentential context:

$$loss_s\left(t, t_r\right) = \alpha\ loss_{cw}\left(t, t_r\right)\ +\ \left(1-\alpha\right) loss_s\left(t, t_r\right) \tag{1}$$

where t is a word n-gram from the corpus and t_r is a corrupted n-gram derived by substituting the centre word of t with a word drawn randomly from the vocabulary. The training objective of C&W and the sentiment-specific training objective are both that the original n-grams obtain higher scores than their corrupted versions by a margin of 1:

$$loss_{cw}\left(t, t_r\right) = max\left(0, 1 - f_{cw}\left(t\right) + f_{cw}\left(t_r\right)\right) \tag{2}$$

$$loss_s\left(t, t_r\right) = max\left(0, 1 - \delta_s\left(t\right) f_s^0\left(t\right) + \delta_s\left(t_r\right) f_s^1\left(t\right)\right) \tag{3}$$

where $f_{cw}\left(t\right)$ is the score of a 2-layer neural network, $f_s^0\left(t\right)$ is the predicted positive score, $f_s^1\left(t\right)$ is the predicted negative score, $s(t)$ is the gold sentiment distribution of an n-gram ([1,0] for positive n-grams and [0,1] for negative ones), and δ_s is an indicator function such that:

$$\delta_s\left(t\right) = \begin{cases} 1 & \text{if } \ s\left(t\right) = [1, 0] \\ -1 & \text{if } \ s\left(t\right) = [0, 1] \end{cases}$$

Although this extension of the C&W model produces embeddings which are able to encode polarity information, it demands very large annotated corpora. In a polarity classification task, a model

trained on 1 million tweets yields an F_1 score of 0.78, whereas a model trained on 12 million tweets obtains an F_1 score of 0.83 (Tang et al. 2014).

Yet there exist only few large annotated corpora. To our knowledge, the largest available emotion-labelled data set is the Hashtag Emotion Corpus, which contains ca. 21,000 tweets. To overcome this problem, Labutov and Lipson (2013) proposed a method that rearranges word representations in their original vector space using a task-specific objective and without directly optimising word embeddings. This alternative approach has multiple advantages: the task-specialisation process of word embeddings is computationally more efficient, the size of the corpus need not be overly large, and pre-trained generic word embeddings can be leveraged making it utilisable as a post-processing step.

3. Methods

3.1 Emotion-specific word representations

To obtain word representations that encode affective orientation and strength, we fine-tune pre-trained embeddings using an emotion classifier.

Model The inputs to our emotion classifier are paragraphs. The words in each paragraph are mapped to their pre-trained representations by an embedding layer. The word vectors are fed to a bidirectional LSTM, followed either by a softmax activation that outputs probability distributions over emotion classes or by a sigmoid activation that produces one probability value for each class. Batch normalisation precedes both the bidirectional LSTM layer and the output layer. In order to obtain specialised representations, the pre-trained word vectors are used to populate the embedding layer of the network and, once optimised, they can function as specialised representations. More details about this encoder can be found in the Appendix.

Motivation for recurrent neural networks Traditional neural networks are not able to make full use of sequential information as they act under the assumption that all inputs occur independently of each other. In contrast, text is sequential in its nature and sentences are not successions of words randomly drawn from a vocabulary; the use of a specific word is dependent on the context of that word.

Recurrent Neural Networks (RNNs) address the issue of sequentiality as they have the potential to represent arbitrarily long context. Nonetheless, not all types of RNNs are appropriate for the analysis of natural language; for example, Simple Recurrent Networks (Elman 1990) are not able to learn long-term dependencies due to vanishing gradients. A variant of recurrent neural networks that is, in principle, capable of making inferences based on long-distance interactions is the Long Short-Term Memory (Hochreiter and Schmidhuber 1997). Long Short-Term Memory (LSTM) networks are designed to overcome the exploding and vanishing gradient problem by enforcing constant error flow and making it possible to recursively exploit information from previous timesteps. From their high accuracy (Linzen et al. 2016) and from further analysis (Gulordava et al. 2018), it seems now clear that LSTMs not only can but do learn about both short and long distance relations in sentences when they are e.g. embedded in a language modelling task.

A further challenge posed by sequences of natural language is that dependence does not flow unidirectionally through sentences. LSTMs, on the contrary, only consider the left context of an input. The following tweet is an example of why it can be important to incorporate the right context of a word or phrase in a sentiment analysis classifier:

(5) My niece calling to sing Happy Birthday to me #love !!

In order to understand the author's effective orientation towards their niece singing Happy Birthday, the affective orientation of *love* should be available at the early timesteps. A bidirectional information flow can be obtained by using two recurrent networks that are presented each sequence forwards

and backwards respectively. Connected to the same output layer, the two networks provide complete sequential information about every time step (Graves and Schmidhuber 2005). This property motivates our use of a bidirectional LSTM as an emotion classifier.

3.2 Lexicon expansion

The lexicon expansion task is defined as follows. We are given a set of emotion classes C, and a set W of word types extracted from a large corpus, which can be partitioned into $L, U \subset W$, the sets of lexicon words and unlabelled words respectively. For ease of notation, we refer to the set cardinalities as follows: $|C| = m$, $|L| = l$, and $|U| = u$. Typically $l \ll u$. We try to find a labelling function λ that maps each unlabelled word to a probability distribution over m classes:

$$\lambda : U \rightarrow \mathbb{R}^m$$
$$w \mapsto (y_1, \ldots, y_m), \quad s.t. \sum_{i=0}^{m} y_i = 1$$

We choose the Label Propagation (LP) algorithm introduced by Zhu and Ghahramani (2002) and propose a novel variant thereof that can solve the lexicon expansion problem. LP is a graph-based semi-supervised technique that propagates labels from labelled to unlabelled nodes through weighted edges. Conceived as an iterative transductive algorithm, LP was shown to have a unique solution. That is, λ can be learned directly, without iteration.

Label propagation More formally, consider a set of label distributions $Y_L = \{y_1, \ldots, y_l\}$ which correspond to the labelled data $L = \{(w_1, y_1), \ldots, (w_l, y_l)\}$, and a matrix $W \in \mathbb{R}^{(l+u)\times(l+u)}$ of pairwise weights. The goal is to estimate the label distribution of the unlabelled data Y_U from the weight matrix W and the gold label distributions Y_L. To do so, we build a fully connected graph using labelled and unlabelled words as nodes. Edges between nodes are defined so that an edge weight w_{ij} is proportional to the geometric proximity of two data points x_i, x_j in high-dimensional space.

In the original version of Label Propagation, weights are defined in terms of squared Euclidean distance and they are controlled by a hyperparameter α:

$$w_{ij} = exp\left(-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)\cdot(\boldsymbol{x}_i - \boldsymbol{x}_j)}{\alpha^2}\right) \tag{4}$$

However, since cosine similarity is commonly preferred as a metric for word embeddings, we define weights as:

$$w_{ij} = \sigma\left(\alpha\left(\frac{\boldsymbol{x}_i \cdot \boldsymbol{x}_j}{||\boldsymbol{x}_i||_2 \; ||\boldsymbol{x}_j||_2}\right) + \beta\right) \tag{5}$$

The use of the logistic function and the bias is motivated by the need (i) to adapt the weight computation to the range of the cosine function and (ii) to obtain a uniform weight formula regardless of the number of parameters.[1]

Finally, define a probabilistic transition matrix $T \in [0,1]^{(l+u)\times(l+u)}$ such that:

$$T_{ij} = P\left(i \rightarrow j\right) = \frac{w_{ij}}{\sum_{k=1}^{l+u} w_{kj}} \tag{6}$$

[1]We also conducted preliminary experiments where we replaced the scalar $\alpha \in \mathbb{R}$ by a vector $\boldsymbol{\alpha} \in \mathbb{R}^d$ that controls edge weights along the d dimensions used to encode nodes. We used the following weighting function: $w_{ij} = \sigma\left(\boldsymbol{\alpha} \cdot \left(\frac{\boldsymbol{x}_i}{||\boldsymbol{x}_i||_2} \odot \frac{\boldsymbol{x}_j}{||\boldsymbol{x}_j||_2}\right) + \beta\right)$, a cosine similarity where every element-wise multiplication ($\odot$ is the Hadamard product) is scaled by a dimension-specific parameter. Unfortunately, this formulation resulted in a loss of performance compared to the one that uses the scalar parameter α.

and a label matrix $Y \in [0,1]^{(l+u)\times m}$, where Y_i stores the probability distribution over labels for node x_i. Notice that T is column-normalised, so that the sum of the probabilities of moving from any node i to node j amounts to 1.

The rows of Y_L are initialised according to the lexicon. If the lexicon only provides one label per word, then a probability value of 1 is assigned to that label. Since the NRC Emotion Lexicon maps words to multiple labels, we uniformly distribute the probability mass among all positively annotated classes. For the initialisation of Y_U, which Zhu and Ghahramani (2002) consider irrelevant, we assign a uniform probability of $1/m$ to every label.

LP algorithm First, the transition probability matrix T needs to be row-normalised. Then, it is partitioned into four sub-matrices:

$$T = \begin{bmatrix} T_{ll} & T_{lu} \\ T_{lu} & T_{uu} \end{bmatrix}$$

Label Propagation consists in iterative updates of the unseen label distributions

$$Y_U \leftarrow T_{uu} Y_U + T_{ul} Y_L \tag{7}$$

and converges to a unique solution (Zhu and Ghahramani 2002):

$$Y_U = (\boldsymbol{I} - T_{uu})^{-1} T_{ul} Y_L \tag{8}$$

As the conditions for computing the matrix inverse are not always met, least square approximation is used in our implementation of label propagation.

Hyperparameters The formulation of edge weights in LP (Equation 4) ensures that when the hyperparameter $\alpha \to 0$, the label of a node is mostly influenced by that of its nearest labelled node, and when $\alpha \to \infty$, the label probability distribution of a node reflects the class frequency in the data since it is affected by virtually all labelled nodes in the graph.

Zhu and Ghahramani (2002) presented two techniques to set the parameter α. The simplest one is to find a minimum spanning tree over all nodes. Kruskal's algorithm (Kruskal 1956) can be used to build a tree whose edges have the property of connecting separate graph components. The second approach relies on gradient descent in order to find the parameter α and β that minimise the entropy H of the predictions.

$$H = -\sum_{ij} Y_{ij} \ \log Y_{ij} \tag{9}$$

After the optimisation of the edge weights, the transition probability matrix T is smoothed via interpolation with a uniform matrix U, such that $U_{ij} = 1 \ / \ (l+u)$, and where ϵ is the interpolation parameter:

$$\tilde{T} = \epsilon U + (1-\epsilon)\, T \tag{10}$$

Since extreme values of α and β can map very similar cosine similarity to the negative tail of the logistic function, where values approach 0 at an exponential rate, the probability matrix needs to be smoothed to avoid that in such cases virtually all the probability mass propagates to one of the analogously near neighbour. Consider, as an example, the parameters $\alpha = 100$ and $\beta = -100$ as well as a word $\boldsymbol{x}_1$ with its nearest neighbours $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$, such that $\cos(\theta(\boldsymbol{x}_1, \boldsymbol{x}_2)) = 0.8$ and $\cos(\theta(\boldsymbol{x}_1, \boldsymbol{x}_3)) = 0.7$. Then, $w_{12} = 2.06\text{e}{-9}$ and $w_{13} = 9.36\text{e}{-14}$. Virtually all probability mass is on w_{12} and labels only propagate from $\boldsymbol{x}_2$ to $\boldsymbol{x}_1$. The probability matrix needs to be smoothed to avoid this problem.

# labels	0	1	2	3	4	5	6	¿ 0
# lemmas	1072	1813	906	447	253	41	2	3462

Table 2: The number of words in the NRC Emotion Lexicon (Mohammad and Turney 2013) having k labels.

Label propagation in batches Label propagation is a computationally efficient algorithm. Since iteration is avoided by directly computing a unique algebraic solution, most computational resources are employed for the calculation of the probabilistic transition matrix T and for the optimisation of the parameters α, β and ϵ.

The size of T can, however, cause memory issues. Consider that the Hashtag Corpus is comprised of $V = 32,930$ word types. LP must thus store the cosine similarity values between each word pair, so $T \in \mathbb{R}^{V \times V}$ and the transition matrix has a size of approximately 2GB for half-precision floating point numbers. While this is still a manageable size, if we wanted to include all the word types for which distributional representations are available from the ENCOW corpus (Section 5.1), T would have a size of ca. 180GB. In other words, an inherent problem of the transition matrix is that it grows quadratically with the vocabulary size.

In order to ensure that LP is tractable for large vocabularies, we introduce Label Propagation in batches. Instead of keeping the entire $T \in R^{V \times V}$ in memory during optimisation, a subset of the vocabulary with size $W < V$ is randomly selected and the corresponding submatrix $T_W \in \mathbb{R}^{W \times W}$ is computed. If enough random submatrices are sampled for optimisation, the obtained parameters will approximate those that would result from optimising on $T \in R^{V \times V}$. Furthermore, the use of random submatrices is motivated by the need of the parameters to learn to adapt to any random subset of the vocabulary.

Randomly selecting W word types can result in high class imbalance: it is possible that a large amount of the reduced vocabulary consists of labelled terms or, conversely, that all words are unlabelled. Both these possibilities contradict the assumption of Label Propagation that $|L| \ll |U|$. Therefore, we fix the distribution of labelled and unlabelled terms to be equal to the proportion they have in the original vocabulary of size V.

4. Linguistic resources

An important attempt to organise affective phenomena was made by Ekman (1992), who introduced the concept of basic emotions, i.e. affective states that seem to share a connection with physiological processes and universal facial expressions. Whether it is possible to identify a fixed number of categories for a multiplicity of moods, attitudes, and traits is outside the scope of this work but we consider a few prominent categorisations to ensure at least a partial psychological grounding of the deployed resources.

Researchers have different opinions as to which affective states constitute a justifiable set of elementary categories. *Ekman's Six* include anger, disgust, fear, joy, sadness, and surprise. Adding trust and anticipation yields Plutchik's eight *primary* emotions (Plutchik 1980). A collection of seven categories (joy, fear, anger, sadness, disgust, shame, and guilt) was used in the ISEAR project

# lemmas	0	1	2	3	4	5	6	¿ 6
# tweets	7513	7207	4187	1442	498	162	30	12

Table 3: The number of tweets in the Hashtag Corpus (Mohammad and Kiritchenko 2015) with k emotion words from the NRC Lexicon (Mohammad and Turney 2013).

Emotion label	# lemmas	# tweets
anger	1555	1247
disgust	761	1058
fear	2816	1476
joy	8240	689
sadness	3830	1191
surprise	3849	534

Table 4: The class frequencies in the NRC Emotion Lexicon (Mohammad and Turney 2013) and in the Hashtag Corpus (Mohammad and Kiritchenko 2015).

(Scherer and Wallbott 1994) whereas Izard (1971) counted nine basic emotions. Mehrabian and Russell (1974) propose a more compact classification including only three independent emotional dimensions—pleasure, arousal and dominance—and many other emotional frameworks have been developed.

While it is possible to choose from a variety of sets of affective states, the amount of currently available emotion-annotated data sets is scarce. The SemEval-2007 Affective Text corpus is one such data set; it is a collection of news titles extracted from newspapers and news web sites (Strapparava and Mihalcea 2007) and it consists of 1250 headlines labelled with Ekman's six emotions. In an attempt to obtain fine-grained labels, six annotators independently assigned to every headline–emotion pair (h, e) an emotion score ranging from 0 to 100 indicating how intensively a news title h conveys the emotion e. The data was originally made available in two sets: a development set consisting of 250 headlines and a test set of 1,000 headlines.[2] In later work on supervised classification methods, the 1,000 news titles were used as training data, and the remaining 250 for testing (Mohammad and Kiritchenko 2015). Moreover, in many experiments, the vector of six emotion scores was made coarser-grained in order to better fit specific types of classification algorithms. In single-label settings, only the most dominant emotion, i.e. the emotion with the highest score, was used as a headline label by Chaffar and Inkpen (2011). For multi-label classification, only emotions with a score higher than a threshold k are considered present in a given headline: while the task description of SemEval-2007 Affective Text indicates $k = 50$, other researchers set the threshold to a lower value (e.g. $k = 25$ was used by Mohammad and Kiritchenko (2015)).

The SemEval dataset does not contain text from social media platforms, which are nowadays a conspicuous source of real language production and increasingly attract interest in the field of opinion mining. An attempt to cover this type of content was made by Mohammad and Kiritchenko (2015), who created a labelled corpus of tweets (Twitter posts) leveraging the use of *hashtags*.[3] Their Hashtag Emotion Corpus consists of about 21,000 tweets annotated with one out of the six emotions proposed by Ekman.

Other relevant Twitter corpora are EmoTweet-28 (Yan et al. 2016), which uses a set of 28 emotions, the multilingual SemEval-2018 Affect in Tweets Dataset (Mohammad et al. 2018)—with 11 emotion classes—and the Twitter corpus collected by Klinger et al. (2018)[4] and annotated with Ekman's Six. Yet other corpora were compiled with sentence-level annotation. Sentences extracted from 22 fairy tales were annotated by (Alm et al. 2005) with 5 emotions (joy, fear, sadness, surprise, and anger-disgust). Aman and Szpakowicz (2007) annotated about 5,000 sentences drawn from blog posts with Ekman's six emotions and a *neutral* category. Neviarouskaya et al. (2009) chose the nine emotion categories proposed by Izard (and a *neutral* one) to label 1,000 sentences extracted from stories on a variety of topics.

[2]This partition reflects the fact that the SemEval-2007 Affective Text task was aimed at unsupervised approaches.

[3]Hashtags are words immediately preceded by a hash symbol which mostly serve to signal the topic of Twitter posts, as well as other sorts of metadata such as the tweeter's mood.

[4]This corpus was not yet available when this study was performed.

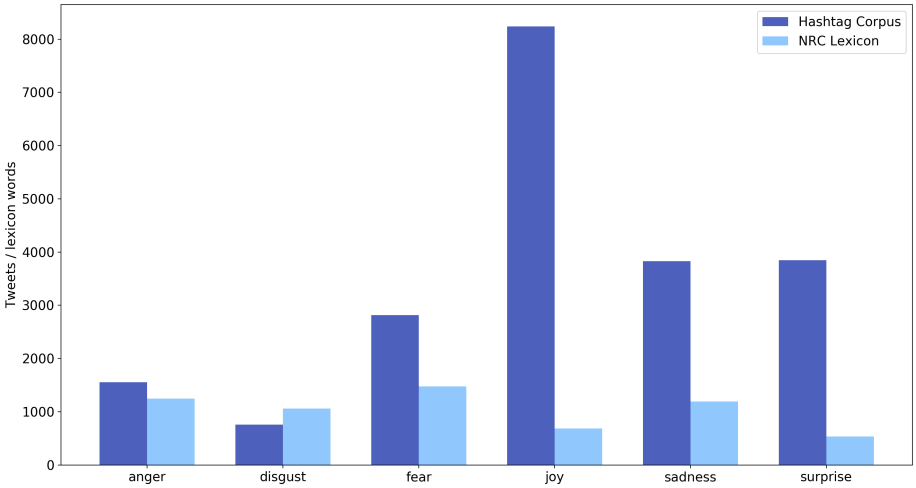

Figure 1: The class distributions of the Hashtag Corpus (Mohammad and Kiritchenko 2015) and the NRC Lexicon (Mohammad and Turney 2013).

Lexical resources are not abundant either. WordNet Affect and the Hashtag Emotion Lexicon use Ekman's Six. WordNet Affect (Strapparava et al. 2004) is a word-emotion association lexicon consisting of a few more than 1,500 terms. The Hashtag Emotion Lexicon (Mohammad and Kiritchenko 2015) contains 11,418 lemmas automatically obtained from the Hashtag Emotion Corpus; each word-emotion pair (w, e) comes with a real-valued Strength of Association (SoA) score between a word w and an emotion e: $SoA(w, e) = PMI(w, e) - PMI(w, \neg e)$.

On the other hand, the NRC Emotion Lexicon (Mohammad and Turney 2013) combines the eight primary emotions proposed by Plutchik with positive and negative polarity, for a total of ten possible labels. The NRC Emotion Lexicon includes ca. 15,000 English words, it was manually created via crowdsourcing, and it uses binary word-emotion association scores.

Mehrabian and Russel's notions of pleasure, arousal, and dominance (1974) were deployed for the manual annotation of Affective Norms for English Words (Bradley and Lang 1999) and, with the addition of *predictability*, for the construction of a multimodal database, annotated using recordings of subjects' physiological responses to affective stimuli (Soleymani et al. 2012). Finally, SentiSense (de Albornoz et al. 2012) and SenticNet 3 (Cambria et al. 2014) make use of WordNet to fine-grain their word annotations.

The Hashtag Emotion Corpus and the NRC Emotion Lexicon

We choose to conduct our experiments using the Hashtag Emotion Corpus and the NRC Emotion Lexicon. The choice of the corpus is led by size considerations, the type of medium it was collected from, and the topics covered by the posts. The largest lexicon available with the same label set would be the Hashtag Emotion Lexicon but it was compiled using the corpus itself. The largest resource with an appropriate label set is then the NRC Emotion lexicon.

The Hashtag Emotion Corpus consists of 21,051 texts annotated with Ekman's six basic emotions and it contains ca. 33000 word types; each text is assigned a single emotion label. The NRC Emotion Lexicon contains 14,182 words. However, only 3,462 lexicon words have at least one of Ekman's six emotion labels—the others are either annotated as *positive*, *negative*, *anticipation*, *trust*, or they are neutral, i.e. no label is set to 1. Each lexicon word is tagged with an average of 0.44 Ekman's emotions. Table 2 reports the labels-per-lemma statistics.

Furthermore, as Figure 1 shows, the class distributions are not uniform. In the lexicon, positive emotions (surprise and joy) are under-represented with respect to negative emotions (Table 4). In the corpus, texts annotated as *joy* form a disproportionately large percentage, while *anger* and *disgust* are the minority classes (Table 4). A text from the Hashtag Corpus contains on average 1.09

Frequency	**lemma**	**Labels**
1218	love	*joy*
621	good	*joy, surprise*
418	afraid	*fear*
411	happy	*joy*
389	friend	*joy*
388	god	*fear, joy*
367	hate	*anger, disgust, fear, sadness*
342	fear	*anger, fear*
311	feeling	*anger, disgust, fear, joy, sadness, surprise*
287	joy	*joy*

Table 5: The 10 most frequent NRC (Mohammad and Turney 2013) lemmas and their emotion labels.

emotion words that also occur in the NRC lexicon and approximately one third of the tweets does not contain any lexicon words. Table 3 presents the distribution of emotion words among texts. An emotion word has an average frequency of 6.64 and only 1,545 lemmas occur at least once in the Hashtag Corpus. Finally, Table 5 illustrates the most frequent emotion words along with their emotion labels according to the NRC Emotion Lexicon.

Overall, the presented statistics show that the coverage of an unexpanded lexicon is too small for emotion classification. These statistics were obtained by lemmatising the Hashtag Corpus and the values resulting from the non-lemmatised corpus do not significantly vary.

5. Evaluation

5.1 Emotion-specific embeddings

The first step of the proposed lexicon expansion method consists in learning emotion-specific word embeddings, i.e. distributed word representations that are able to encode affectual orientation and strength. To learn an emotion-specific vector space we employ a recurrent neural network classifier. The classifier labels tweets from the Hashtag Corpus with Ekman's six basic emotions (Section 4) and uses word vectors as trainable features. As the model learns to classify, we expect word embeddings to store affective orientation.

We use a bidirectional LSTM followed by a softmax or a sigmoid output layer: the softmax activation is used to obtain a probability distribution over the six emotion classes (multinomial classification) whereas a sigmoid layer is used to produce one probability value for each emotion class (multi-label classification). The relevant output of the classifier are the optimised word representations that can be used to compute the spatial similarities necessary for lexicon expansion. The BiLSTM is trained using Keras (Chollet et al. 2015) and a comprehensive list of hyperparameters can be found in Appendix A.1. We experiment with different combinations of three regularisation techniques: ℓ_2 regularisation, batch normalisation, and dropout.

For the initialisation of the embedding layer we rely on a vector space learned from the ENCOW corpus (Schäfer and Bildhauer 2012, Schäfer and DFG 2015) using the CBOW model (Mikolov et al. 2013). The ENCOW dataset contains approximately 425 million sentences and more than 9.5 billion tokens. The chosen vector dimensionality is 300, as suggested in (Mikolov et al. 2013). We

experiment with different thresholds of word frequency, excluding either words whose raw frequency in the corpus is lower than 100 or those occurring less than 150 times. The window size is set to 5 as we expect such a context to be a desirable trade-off between computational complexity and the ability to capture semantic information.

5.2 Emotion lexicon expansion

To expand the NRC Emotion Lexicon we employ our novel variant of the Label Propagation algorithm. Although we have more than 300,000 word vectors at our disposal, label propagation is only applied to the approximately 30,000 vectors that correspond to the word types of the Hashtag Corpus. This decision is motivated by the need to limit the execution time of the propagation algorithm and by the consideration that only the mentioned subset of word embeddings is optimised for emotion-related tasks.

Furthermore, with the proposed batch-based variant of label propagation, we try to approximate the results of standard label propagation optimisation—where the word similarity graph is comprised of all available word types—using multiple batch optimisations—where only a subset of the word types is used to construct the similarity graph. We expect the parameters to learn to robustly adapt to any random subset of the vocabulary and, as a consequence, to discard irrelevant features.

The label propagation hyperparameters $\alpha, \beta \in \mathbb{R}$ are optimised using Tensorflow[5] (Abadi et al. 2015) and their values are reported in Appendix A.2.

5.3 Emotion classification

To test whether a combination of corpus- and lexicon-based approaches improves classification, we use the expanded emotion lexicon to augment pre-trained word representations that serve as features for an emotion classifier. We deploy the same classifier proposed for learning task-specific word representations: an embedding layer maps words to their vector representations, which are the input to a bidirectional LSTM; the output layer can have a softmax or a sigmoid activation for multinomial and multi-label classification respectively.

For the initialisation of the embedding layer, we leverage the vector space previously learned by our bidirectional LSTM (Section 5.1) from the Hashtag Corpus. This 300-dimensional vector space includes approximately 30,000 word types and it represents the corpus-based portion of the information provided to the classifier. To also provide the classifier with lexicon-based cues, we append the label probability distribution vector of a word occurring in the expanded emotion lexicon to the corresponding fine-tuned word vector. Since the lexicon is expanded to all the word types in the Hashtag Corpus, each embedding receives an emotion-specific initialisation. Notice that the label distributions of the original lexicon words are left unvaried due to the properties of Label Propagation.

To ensure that the lexicon is not specialised to the training dataset and to assess its transferability to a different domain, we also classify the SemEval headlines introduced in Section 4 using 1,000 titles for training and the remaining 250 for testing. We train the BiLSTM classifier using Keras and experiment with ℓ_2 regularisation, batch normalisation, and dropout. Precise hyperparameter values are reported in Appendix A.3.

6. Results and discussion

6.1 Emotion lexicon expansion

To perform an intrinsic evaluation of our lexicon expansion method, we choose 10-fold cross-validation. The intersection between the word types of the Hashtag Corpus and the NRC Emotion Lexicon is

[5]The used Tensorflow networks and functions can be found at `github.com/Procope/emo2vec`.

partitioned into 10 equally sized subsamples. The quality of the expanded lexicon is assessed by computing the average Kullback-Leibler divergence between the emotion label probability distributions obtained by normalising the NRC Emotion Lexicon and the distributions resulting from label propagation.

Three baseline lexicon expansion methods are considered: (i) assigning a uniform class distribution to all words, (ii) assigning to all words a distribution where the entire probability mass is given to the majority class according to the Hashtag Corpus, (iii) assigning to all words the overall prior class distribution of the Hashtag Corpus.

Table 6 shows the average Kullback–Leibler divergence for 10-fold cross-validation of the described lexicon expansion techniques. Divergences obtained using label propagation and the uni-

Lexicon expansion	**KL divergence**
Uniform distribution	1.34
Majority class (Hashtag Corpus)	21.32
Prior class distribution (Hashtag Corpus)	1.53
Label propagation ($\alpha \in \mathbb{R}$)	1.31
Batch label propagation ($\alpha \in \mathbb{R}$)	1.31

Table 6: Average Kullback–Leibler divergence for 10-fold cross-validation on the NRC Emotion Lexicon (Mohammad and Turney 2013). The lowest propagation.

form class distribution yield a significantly[6] lower divergence than the prior class distribution of the Hashtag Corpus. Label propagation with a scalar parameter α is the method that best minimises the average Kullback–Leibler divergence. Remarkably, batch label propagation performs similarly to standard propagation although it uses batches of size 5,000 (compared to a total of more than 30,000 word types), showing that batch approximation can work for graph propagation. Yet, although LP reaches the lowest divergence, it is not significantly different from the uniform distribution. It is relevant to note that since KL divergence is computed on the small subset of labelled words, this intrinsic measure overlooks the effect of label propagation on the vast majority of the vocabulary words, for which we think propagation is informative. To explain this surprising result, however, we examine how emotion labels propagate through the graph.

For this purpose, it is useful to consider the trajectories of words belonging to the two discarded emotion categories, trust and anticipation. These words tend to receive the label *joy* more often than the other labels after propagation. This is likely due to the fact that, since joy is the most frequent emotion occurring in the Hashtag Corpus, label propagation tends to assign the label joy more often than the other labels. Interestingly, however, *anticipation* words are labelled 84% of the time as *joy*, *surprise* and *fear* words (in descending frequency). This distribution seems to be a good fit to the definition of anticipation proposed by the Cambridge dictionary: "a feeling of excitement about something that is going to happen in the near future" (anticipation n.d.) Moreover, in a range from interest to distraction, surprise was seen by Plutchik as an opposite to anticipation, the latter mainly consisting of expectation and prediction. It is possible that label propagation, in the lack of a proper label, failed to make a distinction between these two ends of the spectrum and merged them. This behaviour exerts virtually no influence on classification as trust and anticipation tweets (i.e. tweets that contained the hashtags #trust and #anticipation) are not included in the Hashtag Emotion Corpus but it exemplifies the dynamics of label propagation and it gives us hope that they will still be useful in a downstream task.

[6]We used Approximate Randomization Tests (Edgington 1969) adjusting the significance value 0.05 using Šidák correction for multiple tests (Šidák 1967).

6.2 Emotion classification

To extrinsically evaluate the proposed emotion classifier, we compare it with (Mohammad and Kiritchenko 2015) and with two baselines. The first one is a version of the bidirectional LSTM classifier introduced at the beginning of this section that only exploits our emotion-specific word embeddings. As an alternative, the pre-trained emotion-specific embeddings are concatenated with the probability distributions indicated by the unexpanded NRC Emotion Lexicon. Words that do not occur in the lexicon have their vectors concatenated with a vector of probabilities from a uniform distribution. The uniform distribution outperforms the overall probability distribution of emotions in the lexicon as an initialisation method.

The precision, accuracy, and F_1 score of all the presented classifiers are reported in Tables 7 and 8. All metrics are computed using micro-averaging to allow comparisons with previous work and to reduce the effect of label imbalance. Using macro-averaging would assign more weight to the majority classes, for which classifiers tend to perform better due to the larger amount of training instances.

Classifier	$\boldsymbol{P}$	$\boldsymbol{R}$	$\boldsymbol{F_1}$
One-vs-all SVM (Mohammad and Kiritchenko 2015)	55.1	45.6	49.9
Multinomial BiLSTM	55.0	55.0	55.0
Multinomial BiLSTM + NRC Emotion Lexicon	55.2	55.2	55.2
Multinomial BiLSTM + expanded lexicon ($\alpha \in \mathbb{R}$)	**56.2**	**56.2**	**56.2**
Students	40.9	40.4	40.6

Table 7: Results of classification on the Hashtag Emotion Corpus: precision, recall, and F_1 score. The expanded lexicon provides an absolute improvement of 6.3 in micro-averaged F_1 score compared to (Mohammad and Kiritchenko 2015).

For the Hashtag Emotion Corpus, the bidirectional LSTM classifier introduced as a baseline outperforms the one-vs-all SVM with binary features, setting a relatively high lower bound for our task. Including as features the label distributions of the NRC Emotion Lexicon slightly increases classification accuracy, suggesting that corpus- and lexicon-based clues are complementary. The limited increment in accuracy can be explained by the fact that a text from the Hashtag Corpus includes on average 1.09 NRC emotion words and that approximately one third of the tweets does not contain any NRC lemmas.

In contrast, the LSTM classifier shows a remarkable increase in accuracy when the expanded lexicon is provided. Although the quality of the expanded lexicon is lower than the quality of the hand-annotated NRC Emotion Lexicon, the wider coverage of the former seems to successfully help the classifier. Regardless of whether the LSTM classifier uses no lexicon, the NRC lexicon, or the expanded one, the multinomial variant consistently obtains higher accuracy scores. The classification report of our best classifier is presented in Table 9.

In the evaluation on the Hashtag Emotion Corpus, the pre-training of sentiment embeddings on the Hashtag Emotion Corpus may positively influence the ten-fold cross-validation on that same corpus.[7] Unfortunately, ten-fold cross validation of the complete system was too expensive to perform using the resources available. This posed us to do a separate evaluation on the SemEval headlines data set, which is disjoint and thus does not have this problem. Despite this shortcoming in the evaluation on the Hashtag Emotion Corpus, the results show us that the BiLSTM models (which all have this advantage) benefit strongly from an expanded lexicon.

Humans as classifiers In Section 2.1, we have reported the inter-annotator agreement studies conducted by Strapparava and Mihalcea (2007) for the SemEval headlines corpus. These have shown that trained annotators agree with a Pearson correlation of 53 using a simple average over classes; a

[7] As was pointed out by one of the reviewers.

Classifier	P	R	F_1
One-vs-all SVM (Mohammad and Kiritchenko 2015)			
1. n-grams in headlines dataset and Hashtag Corpus + domain adaptation	46.0	35.5	40.1
2. n-grams in headlines dataset + NRC Emotion Lexicon	**46.7**	38.6	42.2
Multi-label BiLSTM	38.8	50.3	43.8
Multi-label BiLSTM + NRC Emotion Lexicon	39.2	**50.9**	44.3
Multi-label BiLSTM + expanded lexicon	43.1	48.9	**45.9**

Table 8: Results of classification on the SemEval headlines dataset (Strapparava and Mihalcea 2007): precision, recall, and F_1 score. The bidirectional LSTM informed with the expanded lexicon is the best classifier when it comes to combining precision and recall.

frequency-based average yields a correlation of 43. Rather than reproducing another inter-annotator agreement study, we test the accuracy of an untrained person with respect to an emotion-annotated data set, the Hashtag Corpus.

Our survey includes 33 participants; these undergraduate and graduate students were asked to read 25 tweets and to assign to each of them one of Ekman's basic emotions. Each participant was given a different set of tweets, for a total of 825 collectively classified instances. All participants were enrolled in an international study program, have a good command of English but various L1s.[8] The results of our survey are shown in Table 10.

In the attempt to establish an upper bound, we find evidence that an untrained annotator is considerably less accurate than our LSTM classifiers. This outcome suggests that the accuracy of emotion classifiers is necessarily limited by the lack of contextualisation and background information about the author of a tweet. Assigning an emotion to a short paragraph is a hard task for both a human and a statistical classifier as it requires more information than it is available in the paragraph itself.

Emotion	P	R	F_1
anger	38	27	32
disgust	40	18	25
fear	58	52	55
joy	66	76	71
sadness	40	44	42
surprise	53	46	49
average	56.2	56.2	56.2

Table 9: Results of the best performing classifier (BiLSTM with expanded lexicon) on the Hashtag Corpus.

Emotion	P	R	F_1
anger	25	50	33
disgust	18	70	29
fear	48	22	30
joy	52	46	49
sadness	50	52	51
surprise	40	23	29
average	40.9	40.4	40.6

Table 10: Precision, recall, and F_1 score of student answers to a survey based on the Hashtag Corpus.

A possible explanation for the surprisingly low accuracy of the students is to be found in the quality of the Hashtag Corpus. Mohammad and Kiritchenko (2015) clarify that there are essentially three types of tweets: those where the affectual orientation is straightforward even without the emotion hashtag, those where only the hashtag makes the affectual orientation explicit, and those where text and hashtag seem to conflict. In the second and in the third case, humans may tend to answer according to a uniform prior while the model's prior belief corresponds to the class distribution of

[8] As noted by one of the reviewer, different levels of language proficiency can affect lexical knowledge. We acknowledge that a group formed exclusively by native English speakers would have given more robustness to the survey but it was not possible to gather such a group during this study.

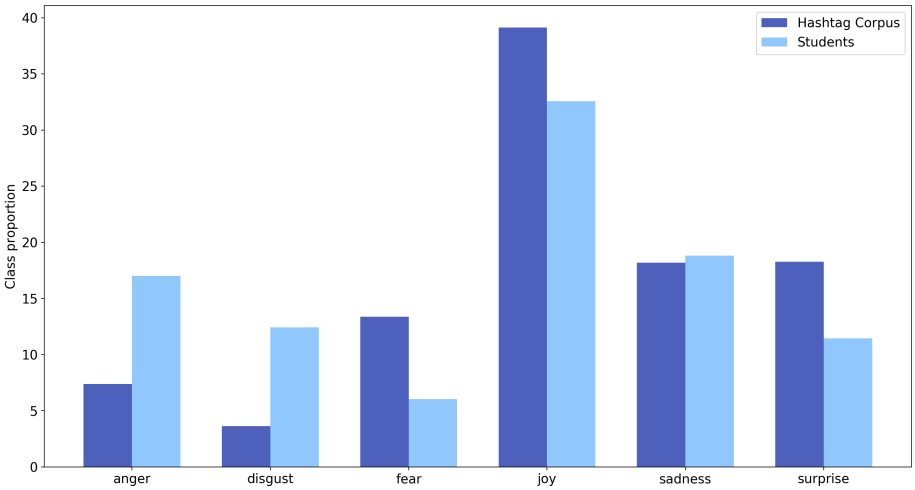

Figure 2: The class distributions in the Hashtag Corpus (Mohammad and Kiritchenko 2015) and in the answers to the survey (true positives + false positives). The values on the y-axis are expressed in percentage form.

the dataset. The accuracy of the model seems to benefit from such an informed prior. Interestingly, Figure 2 shows how the—broadly speaking—posterior class distribution of students coarsely corresponds to the data likelihood. Moreover, the different levels of English proficiency among the participants have likely affected their ability to recognise the prevalent affect of texts, especially when this is expressed using infrequent words which might be unknown to non-native readers.

A final consideration is that, in the absence of clear emotional content, both humans and the model are expected to comply with the gold standard label selected by annotators whose agreement score is remarkably low. It is likely that different individuals adopt varying decision strategies and thresholds to determine the emotion conveyed by a piece of text. It is worth exploring, in future, whether the lexicon expansion algorithm or the downstream classifier can approximately capture similar reasoning processes.

7. Conclusion

In this paper, we have argued and shown that combining corpus-based and lexicon-based approaches to affect detection can improve the accuracy of emotion classifiers. In particular, we have shown that label propagation can be applied to a fully connected graph wherein words are vertices and edges are weighted by geometric proximity in order to expand an existing emotion lexicon. We have additionally demonstrated that graph propagation can rely on task-specific word embeddings to initialise node representations.

We have introduced two variations of the Label Propagation algorithm: (i) a novel formulation of the transition edge weights that allows the use of cosine similarity as a distance metric for word representations, (ii) and a batch-based training algorithm whose reduced time and memory requirements make expansion tractable for large vocabularies.

To assess the utility of lexicon expansion, we have provided a bidirectional LSTM classifier with the class probability distributions inferred through label propagation—i.e. with the extended emotion lexicon. Our comparisons with the model of Mohammad and Kiritchenko (2015), a bidirectional LSTM, and a bidirectional LSTM using the NRC Lexicon have shown that the model that uses the expanded emotion lexicon consistently outperforms the other models.

In future, in light of the latest developments in the field of neural approaches to graph processing, we plan to perform label propagation using Graph Convolutional Networks (Defferrard et al. 2016, Kipf and Welling 2016), which have been shown to combine high classification accuracy on various benchmark graphs with fast training times.

We also plan to employ (i) GloVe embeddings (Pennington et al. 2014) as an initialisation for trainable specialised word representations since they were shown to capture semantic similarity better than Word2Vec embeddings as well as (ii) the morphology-aware FastText vectors (Bojanowski et al. 2016), which obtain high scores on various affect analysis benchmarks. Although emotion-specific embeddings have proven to be a reliable source for the construction of similarity graphs and they can be enhanced introducing e.g. lexical-contrast from semantic networks, there exist alternative representations: we will experiment with character-based models and with advanced co-occurrence statistics.

References

Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng (2015), TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org. http://tensorflow.org/.

Alm, Cecilia Ovesdotter, Dan Roth, and Richard Sproat (2005), Emotions from Text: Machine Learning for Text-based Emotion Prediction, *Proceedings of the conference on human language technology and empirical methods in natural language processing*, Association for Computational Linguistics, pp. 579–586.

Aman, Saima and Stan Szpakowicz (2007), Identifying Expressions of Emotion in Text, *International Conference on Text, Speech and Dialogue*, Springer, pp. 196–205.

Andreevskaia, Alina and Sabine Bergler (2008), When Specialists and Generalists Work Together: Overcoming Domain Dependence in Sentiment Tagging, *ACL*, pp. 290–298.

anticipation (n.d.), *Cambridge English Dictionary*, Cambridge University Press. Retrieved from https://dictionary.cambridge.org/dictionary/english/anticipation.

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov (2016), Enriching Word Vectors with Subword Information, *CoRR*. http://arxiv.org/abs/1607.04606.

Bradley, Margaret M and Peter J Lang (1999), Affective Norms for English Words (ANEW): Instruction manual and affective ratings, *Technical report*, Citeseer.

Bravo-Marquez, Felipe, Eibe Frank, Saif M Mohammad, and Bernhard Pfahringer (2016), Determining Word–Emotion Associations from Tweets by Multi-Label Classification, *WI'16*, IEEE Computer Society, pp. 536–539.

Cambria, Erik, Daniel Olsher, and Dheeraj Rajagopal (2014), SenticNet 3: a Common and common-Sense knowledge base for cognition-driven sentiment analysis, *Twenty-eighth AAAI Conference on Artificial Intelligence*.

Chaffar, Soumaya and Diana Inkpen (2011), Using a Heterogeneous Dataset for Emotion Analysis in Text, *Canadian Conference on Artificial Intelligence*, Springer, pp. 62–67.

Chollet, François et al. (2015), Keras.

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa (2011), Natural Language Processing (Almost) from Scratch, *Journal of Machine Learning Research* **12** (Aug), pp. 2493–2537.

de Albornoz, Jorge Carrillo, Laura Plaza, and Pablo Gervás (2012), SentiSense: An Easily Scalable Concept-Based Affective Lexicon for Sentiment Analysis, *Proceedings of the 8th International Conference on Language Resources and Evaluation. LREC*, pp. 3562–3567.

Defferrard, Michaël, Xavier Bresson, and Pierre Vandergheynst (2016), Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering, *CoRR*. http://arxiv.org/abs/1606.09375.

Edgington, Eugene S (1969), Approximate Randomization Tests, *The Journal of Psychology* **72** (2), pp. 143–149, Taylor & Francis.

Ekman, Paul (1992), An Argument For Basic Emotions, *Cognition & emotion* **6** (3-4), pp. 169–200, Taylor & Francis.

Elman, Jeffrey L (1990), Finding Structure in Time, *Cognitive science* **14** (2), pp. 179–211, Wiley Online Library.

Esuli, Andrea and Fabrizio Sebastiani (2007), SentiWordNet: A High-Coverage Lexical Resource for Opinion Mining, *Evaluation* pp. 1–26.

Graves, Alex and Jürgen Schmidhuber (2005), Framewise Phoneme Classification with Bidirectional LSTM and other Neural Network Architectures, *Neural Networks* **18** (5), pp. 602–610, Elsevier.

Gulordava, Kristina, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni (2018), Colorless Green Recurrent Networks Dream Hierarchically, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Vol. 1, pp. 1195–1205.

Hochreiter, Sepp and Jürgen Schmidhuber (1997), Long Short-Term Memory, *Neural computation* **9** (8), pp. 1735–1780, MIT Press.

Izard, Carroll E (1971), The Face of Emotion, Appleton-Century-Crofts.

Joachims, Thorsten (1999), Transductive Inference for Text Classification using Support Vector Machines, *ICML*, Vol. 99, pp. 200–209.

Kennedy, Alistair and Diana Inkpen (2006), Sentiment Classification of Movie Reviews using Contextual Valence Shifters, *Computational intelligence* **22** (2), pp. 110–125.

Kipf, Thomas N. and Max Welling (2016), Semi-Supervised Classification with Graph Convolutional Networks, *CoRR*. http://arxiv.org/abs/1609.02907.

Klinger, Roman, Orphée De Clercq, Saif M Mohammad, and Alexandra Balahur (2018), IEST: WASSA-2018 Umplicit Emotions Shared Task, *Proceedings of The International Conference on Language Resources and Evaluation*.

Kruskal, Joseph B (1956), On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem, *Proceedings of the American Mathematical society* **7** (1), pp. 48–50, JSTOR.

Labutov, Igor and Hod Lipson (2013), Re-embedding Words, *ACL (2)*, pp. 489–493.

Linzen, Tal, Emmanuel Dupoux, and Yoav Goldberg (2016), Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies, *Transactions of the Association for Computational Linguistics* **4**, pp. 521–535. https://transacl.org/ojs/index.php/tacl/article/view/972.

Mehrabian, Albert and James A Russell (1974), *An Approach to Environmental Psychology*, The MIT Press.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean (2013), Efficient Estimation of Word Representations in Vector Space, *arXiv preprint arXiv:1301.3781*.

Mohammad, Saif, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko (2018), SemEval-2018 Task 1: Affect in Tweets, *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 1–17.

Mohammad, Saif M (2015), Sentiment Analysis: Detecting Valence, Emotions, and other Affectual States from Text, *Emotion Measurement* pp. 201–238.

Mohammad, Saif M and Peter D Turney (2013), Crowdsourcing a Word–Emotion Association Lexicon, *Computational Intelligence* **29** (3), pp. 436–465, Wiley Online Library.

Mohammad, Saif M and Svetlana Kiritchenko (2015), Using Hashtags to Capture Fine Emotion Categories from Tweets, *Computational Intelligence* **31** (2), pp. 301–326, Wiley Online Library.

Neviarouskaya, Alena, Helmut Prendinger, and Mitsuru Ishizuka (2009), Compositionality Principle in Recognition of Fine-Grained Emotions from Text, *ICWSM*.

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan (2002), Thumbs up? Sentiment Classification using Machine Learning Techniques, *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, Association for Computational Linguistics, pp. 79–86.

Pang, Bo, Lillian Lee, et al. (2008), Opinion Mining and Sentiment Analysis, *Foundations and Trends® in Information Retrieval* **2** (1–2), pp. 1–135, Now Publishers, Inc.

Pennington, Jeffrey, Richard Socher, and Christopher D Manning (2014), Glove: Global Vectors for Word Representation, *EMNLP*, Vol. 14, pp. 1532–1543.

Petrovic, Sasa, Miles Osborne, and Victor Lavrenko (2010), The Edinburgh Twitter Corpus, *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, pp. 25–26.

Plutchik, Robert (1980), A General Psychoevolutionary Theory of Emotion, *Theories of emotion* **1** (3-31), pp. 4, Academic Press New York.

Qiu, Guang, Bing Liu, Jiajun Bu, and Chun Chen (2009), Expanding Domain Sentiment Lexicon through Double Propagation, *IJCAI*, Vol. 9, pp. 1199–1204.

Schäfer, Roland and Felix Bildhauer (2012), Building Large Corpora from the Web Using a New Efficient Tool Chain, *LREC*, pp. 486–493.

Schäfer, Roland and Linguistic Web Characterization DFG (2015), Processing and Querying Large Web Corpora with the COW14 Architecture, *Proceedings of the 3rd Workshop on Challenges in the Management of Large Corpora*, pp. 28–34.

Scherer, Klaus R and Harald G Wallbott (1994), Evidence for Universality and Cultural Variation of Differential Emotion Response Patterning, *Journal of personality and social psychology* **66** (2), pp. 310, American Psychological Association.

Šidák, Zbyněk (1967), Rectangular Confidence Regions for the Means of Multivariate Normal Distributions, *Journal of the American Statistical Association* **62** (318), pp. 626–633, Taylor & Francis.

Soleymani, Mohammad, Jeroen Lichtenauer, Thierry Pun, and Maja Pantic (2012), A Multimodal Database for Affect Recognition and Implicit Tagging, *IEEE Transactions on Affective Computing* **3** (1), pp. 42–55, IEEE.

Srivastava, Nitish, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov (2014), Dropout: a Simple Way to Prevent Neural Networks from Overfitting, *Journal of Machine Learning Research* **15** (1), pp. 1929–1958.

Strapparava, Carlo, Alessandro Valitutti, et al. (2004), WordNet Affect: an Affective Extension of WordNet, *LREC*, Vol. 4, Citeseer, pp. 1083–1086.

Strapparava, Carlo and Rada Mihalcea (2007), SemEval-2007 Task 14: Affective Text, *Proceedings of the 4th International Workshop on Semantic Evaluations*, Association for Computational Linguistics, pp. 70–74.

Strapparava, Carlo and Rada Mihalcea (2008), Learning to Identify Emotions in Text, *Proceedings of the 2008 ACM symposium on Applied computing*, ACM, pp. 1556–1560.

Taboada, Maite, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede (2011), Lexicon-based Methods for Sentiment Analysis, *Computational linguistics* **37** (2), pp. 267–307, MIT Press.

Tang, Duyu, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin (2014), Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification, *ACL (1)*, pp. 1555–1565.

Vapnik, Vladimir Naumovich and Vlamimir Vapnik (1998), *Statistical Learning Theory*, Vol. 1, Wiley New York.

Yan, Jasy Liew Suet, Howard R Turtle, and Elizabeth D Liddy (2016), EmoTweet-28: a Fine-Grained Emotion Corpus for Sentiment Analysis, *Proceedings of the 10th International Conference on Language Resources and Evaluation. LREC*, pp. 1149–1156.

Yang, Min, Wenting Tu, Ziyu Lu, Wenpeng Yin, and Kam-Pui Chow (2015), LCCT: a Semi-supervised Model for Sentiment Classification, *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, Association for Computational Linguistics (ACL).

Zhou, Zhixin, Xiuzhen Zhang, and Mark Sanderson (2014), Sentiment Analysis on Twitter through Topic-based Lexicon Expansion, *Australasian Database Conference*, Springer, pp. 98–109.

Zhu, Xiaojin and Zoubin Ghahramani (2002), Learning from Labeled and Unlabeled Data with Label Propagation, Citeseer.

Appendix A. Hyperparameters

The software related to this paper is open-source and available at `github.com/Procope/emo2vec`.

A.1 Emotion-specific embeddings

The emotion classifier was trained using Keras (Chollet et al. 2015). Here, we provide an overview of the hyperparameters that we used.

- **Solver**: Adagrad, with a learning rate decay of 1e−4.
- **Learning rate**: The initial learning rate was set to 0.005.
- **Epochs**: The model was trained for 30 epochs.
- **Dimensionality**: We used 300-dimensional word representations with a word frequency threshold of 150.
- **LSTM layer**: The forward and backward layers were trained with 128 output dimensions. Increasing the number of output dimensions did not provide an improvement.
- **Regularisation**: 10% dropout and 20% recurrent dropout (Srivastava et al. 2014); a stronger dropout did not yield a better performance. Additionally, we applied ℓ_2 regularisation (Keras default).

A.2 Label propagation

The following parameters of our variant of Label Propagation were optimised using Tensorflow (Abadi et al. 2015).

- **Standard label propagation**
  - epochs: 100
  - $\alpha = 0.007$
  - $\beta = 2.41$
  - $d = 300$
- **Batch-based label propagation**
  - batch size: 5000
  - number of batches: 1000
  - epochs per batch: 3
  - $\alpha = -0.001$
  - $\beta = 0.9$
  - $d = 300$

A.3 Emotion classification

This is an of overview the hyperparameters that were used for the best emotion classifier.

- **Solver**: Adagrad, with a learning rate decay of 1e−3.
- **Learning rate**: The initial learning rate was set to 0.01.

- **Epochs**: The model was trained for 20 epochs.
- **Dimensionality**: We used 300-dimensional word representations with a word frequency threshold of 150.
- **LSTM layer**: The forward and backward layers were trained with 128 output dimensions. Increasing the number of output dimensions did not provide an improvement.
- **Regularisation**: 10% dropout and 20% recurrent dropout (Srivastava et al. 2014); a stronger dropout did not provide better performance. Additionally, we applied ℓ_2 regularisation (Keras default).