Learning Emotion-enriched Word Representations

Ameeta Agrawal, Aijun An and **Manos Papagelis**
Department of Electrical Engineering and Computer Science
York University, Toronto, Canada
{ameeta,aan,papaggel}@eecs.yorku.ca

Abstract

Most word representation learning methods are based on the distributional hypothesis in linguistics, according to which words that are used and occur in the same contexts tend to possess similar meanings. As a consequence, emotionally dissimilar words, such as "*happy*" and "*sad*" occurring in similar contexts would purport more similar meaning than emotionally similar words, such as "*happy*" and "*joy*". This complication leads to rather undesirable outcome in predictive tasks that relate to affect (emotional state), such as emotion classification and emotion similarity. In order to address this limitation, we propose a novel method of obtaining emotion-enriched word representations, which projects emotionally similar words into neighboring spaces and emotionally dissimilar ones far apart. The proposed approach leverages distant supervision to automatically obtain a large training dataset of text documents and two recurrent neural network architectures for learning the emotion-enriched representations. Through extensive evaluation on two tasks, including emotion classification and emotion similarity, we demonstrate that the proposed representations outperform several competitive general-purpose and affective word representations.

1 Introduction

Emotion detection from text is the task of identifying emotions from natural language data such as reviews, blogs, news articles, and so on (Alm et al., 2005; Aman and Szpakowicz, 2007). While numerous taxonomies of emotions have been proposed (Ekman, 1992; Plutchik, 1980; Parrott, 2001), most psychologists agree that an emotion is a feeling that characterizes the state of mind such as *happiness*, *sadness*, *anger*, among others. The ability to detect emotions in text is critical for a number of applications and services in diverse domains, including market research, customer relations, gaming, and intelligent tutoring systems, to name a few (Mohammad and Turney, 2013).

Despite its potentially wide-spread use, the automatic detection of emotions remains a challenging multi-class, sometimes multi-label, classification problem due to a number of reasons, including: (i) different emotion models consist of different number and types of emotion categories; (ii) emotions are not only subjective but also fuzzy, with more than one emotion occurring at the same time. As a result, development of emotion related resources, such as training data, has been limited to a few manually annotated datasets or lexicons, a process that requires much time and effort, and is expensive.

To solve the limited training data problem, the recent success of word embeddings has garnered increased attention in the design of emotion classification systems (Bravo-Marquez et al., 2016; Pool and Nissim, 2016; Mohammad and Bravo-Marquez, 2017). Word embeddings are distributed word representations (Collobert et al., 2011; Turian et al., 2010), where each word w in the vocabulary $\mathcal{V}$ is mapped into a dense, low-dimensional, continuous-valued vector $x \in \mathbb{R}^d$, $d \ll |\mathcal{V}|$. The underlying idea is that words that frequently occur together in same contexts get mapped to similar regions of the vector space.

Most embeddings (Mikolov et al., 2013b; Pennington et al., 2014) are typically modeled using the syntactic context of words following the distributional hypothesis, i.e., words which occur in similar

Proceedings of the 27th International Conference on Computational Linguistics, pages 950–961
Santa Fe, New Mexico, USA, August 20-26, 2018.

word pair	GloVe	CBOW
(happy, joy) ↑	0.601	0.355
(happy, sad) ↓	0.643	0.535
(cry, weep) ↑	0.605	0.574
(cry, laugh) ↓	0.657	0.403

Table 1: Cosine similarity between emotionally similar (↑) and emotionally dissimilar (↓) word pairs

contexts tend to be semantically similar. While the property of semantic similarity is beneficial in a number of tasks, modeling emotionally *dissimilar* words with similar contexts into neighboring spaces becomes counterproductive in affective tasks such as emotion classification. To further motivate this limitation, Table 1 presents the cosine similarity between the word vectors of a few word pairs obtained from popular pre-trained word embeddings such as GloVe (Pennington et al., 2014) and CBOW (Mikolov et al., 2013b). According to the similarity scores, both GloVe and CBOW rate the word pair (*happy*, *sad*) as more similar than (*happy*, *joy*).

The effectiveness of word embeddings has been shown to be task-dependent (Labutov and Lipson, 2013; Bansal et al., 2014) and while there is some work on generating task-specific embeddings (Kalchbrenner et al., 2014; Tang et al., 2014; Chen and Manning, 2014; Qu et al., 2015), there is little work specifically exploring the role of task-specific *emotion-enriched* embeddings.

In this paper, we propose learning emotion-enriched word representations[1], which we call Emotion Word Embeddings (EWE), in order to project emotionally similar words into neighboring spaces. Towards that end, first, a method of distant supervision is employed to automatically create a large training dataset with a rich spectrum of emotions. Then, two recurrent neural network architectures are employed to learn emotion-aware word representations by leveraging noisy, but large training data. Specifically, we use Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) to capture the *semantic* information between the words of the text document as well as the *emotion* information provided in the form of the target label obtained through distant supervision. Experimental evaluation demonstrates the effectiveness of learned emotion embeddings in the two tasks of emotion classification and emotion similarity.

The major contributions of this work include: (i) a novel distant supervision method for automatically labeling a large corpus of training data with fine-grained emotions; (ii) two LSTM model architectures for learning emotion-enriched word embeddings from text documents (a single-label model and a multi-label model); (iii) and, an extensive evaluation of the learned word vectors on two tasks: emotion classification over four domains of text (blogs, fairy tales, personal experiences, and tweets) and emotion similarity.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 describes the proposed model, followed by the experimental setup and results in Section 4. Section 5 presents the qualitative analysis and finally, Section 6 concludes the paper.

2 Related Work

There exists a large body of work discussing representation learning. Generic word vector models use unannotated text to learn the embedding vector of each term as a fixed length continuous representation by predicting adjacent terms in the document (Bengio et al., 2003; Collobert and Weston, 2008; Collobert et al., 2011; Mikolov et al., 2013b; Mikolov et al., 2013a; Pennington et al., 2014). Incorporating distributed word embeddings as features has proven effective in a variety of natural language processing tasks, including parsing (Socher et al., 2013), language modeling (Bengio et al., 2003; Mnih and Hinton, 2008) and sentiment analysis (Socher et al., 2011; Labutov and Lipson, 2013; Tang et al., 2014; Tang et al., 2016). However, the effectiveness of generic word embeddings has been shown to be heavily task-dependent (Labutov and Lipson, 2013; Bansal et al., 2014).

[1]Available for download: `https://www.dropbox.com/s/5egqnbktbfxp2im/ewe_uni.txt.zip?dl=0`

To increase the effectiveness of generic word embeddings, therefore, there have been some lines of work in using neural networks for inducing task-specific *affective* embeddings. Socher et al. (2011) learned vector space representations for multi-word phrases using recursive autoencoders for the task of sentiment analysis. Labutov and Lipson (2013) produced task-specific embeddings from existing word embeddings for sentiment analysis. Kalchbrenner et al. (2014) trained their models on a large dataset of tweets, where a tweet was automatically labeled as positive or negative depending on the emoticon that occurs in it. Tang et al. (2014; 2016) also induced embeddings from scratch for sentiment analysis using a dataset of 10M tweets obtained through distant supervision labeled with positive and negative emoticons. More recently, affective word representations have been obtained using a corpus of almost 1B tweets weakly labeled with a set of 64 emojis (Felbo et al., 2017). An alternative to learning task-specific embeddings from scratch or updating existing embeddings using neural networks is post-processing (or fine-tuning) existing embeddings with respect to some external knowledge source such as a lexicon (Faruqui et al., 2015).

All the above-mentioned approaches of learning task-specific affective embeddings (Tang et al., 2014; Tang et al., 2016; Felbo et al., 2017) rely on tweets data obtained from Twitter, automatically labeled using emoticons. However, tweets data do not generalize well to texts from other domains such as blogs, narratives, etc. Instead, we explore a novel domain of text (product reviews) to present a more generalizable approach to obtaining large-scale training data using distant supervision. In addition, while previous embeddings were trained on corpora of sizes ranging from 10M to 1B tweets, our models are able to learn rich representations from a much smaller dataset of about 200K reviews. Furthermore, although a binary spectrum of positive and negative sentiment (Tang et al., 2014) or a large axis of 64 emojis (Felbo et al., 2017) has been previously used to generate representations, we align our embeddings along an emotion model firmly grounded in psychology which remains unexplored yet. Lastly, while the previous approaches used only a single-label setting (i.e., only one affect label per document), we propose modeling a more natural multi-label setting where a document can be associated with more than one emotion label.

3 Emotion-enriched Word Representations

In this section, we first describe two neural network models and their components for learning Emotion Word Embeddings (EWE). Then, we describe the process of automatically obtaining a large training dataset of text documents labeled with emotions through distant supervision.

3.1 Training Word Embeddings using LSTM

Let $\mathcal{V} = \left\{w_1, ..., w_{|\mathcal{V}|}\right\}$ be the vocabulary of word tokens in the annotated dataset. Each word w_i is represented as a n-dimensional continuous vector $\mathbf{x}_i \in \mathbb{R}^n$ and the full embedding matrix is $E \in \mathbb{R}^{n \times |\mathcal{V}|}$. Starting from original embeddings $\mathbf{x}_i^o$ of word w_i (initialized either randomly or through some pre-trained word embeddings), the goal is to learn emotion-enriched embeddings $\mathbf{x}_i^e$ for w_i.

The LSTM (Long Short-Term Memory) model finds a dense low dimensional representation of words by sequentially and recurrently processing each word in a document. Specifically, the inputs of the LSTM are preprocessed text documents that consist of a sequence of words and their corresponding target variable. Let $\mathcal{D} = \{(d_1, y_1), ..., (d_D, y_D)\}$ denote an annotated dataset of documents, where $d = \{w_1, w_2, ..., w_N\}$ denotes a text document consisting of a sequence of N words and y_i is the corresponding emotion label distribution for document d_i. The words of the text document are, first, converted into vector representations, which are then sequentially fed into the LSTM model left-to-right. Then, through back-propagation, the original word vectors get updated during training, producing emotion-enriched embeddings $\mathbf{x}_i^e$ for all $w_i \in \mathcal{V}$.

In this work, we consider two model architectures to capture the *context* information by modeling the long-range dependencies between the words of a text document and *emotion* information provided through the target label to map each word into an affective embedding space. Model 1 (EWE_{UNI}) considers a single emotion label for each document, whereas Model 2 ($\text{EWE}_{\text{MULTI}}$) allows multiple labels for a document. Figure 1 presents an overview of the proposed framework. First, we create a cor-

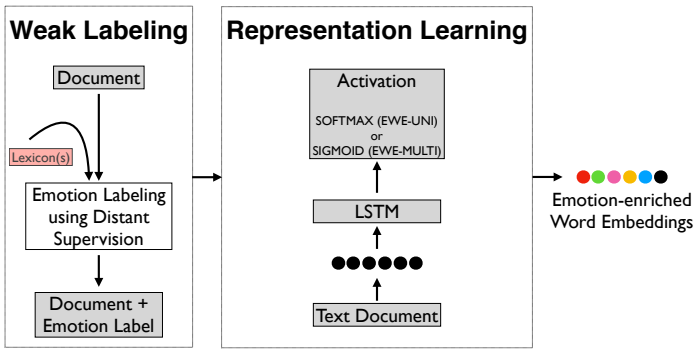


Figure 1: Overview of the framework for obtaining emotion-aware word representations

pus of emotion-labeled documents using emotion lexicons through a distant supervision process (to be described in Section 3.2). Then this corpus is used as training data to learn emotion-enriched word representations using LSTM. In other words, while document-level (entire examples) labeling is used to create the training set, the embeddings get updated at individual word level.

3.1.1 Model 1: EWE_{UNI}

Most words evoke only one emotion depending on the context. As an example, consider two benchmark emotion datasets (Alm et al., 2005; Aman and Szpakowicz, 2007) where each sentence is annotated with a single emotion label. Guided by this intuition, we propose EWE_{UNI} which follows a multi-class setting, where there exists only one valid mutually exclusive emotion label l_i for a text document d_i, and $l_i \in \mathcal{L}$, where $\mathcal{L} = \{l_1, ..., l_k\}$ denotes a discrete, finite set of k emotions.

Given an annotated document with its associated emotion label, the target value y is a one-hot vector, where the values of all the indices but one are 0. For example, if d is labeled with emotion l_i, then it holds that:

$$y_j = \begin{cases} 1, & \text{if } y_j = l_i \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

The neural network consists of one hidden layer, with the embedding matrix E added to the input layer. To predict the emotion label of the input text, an output layer with a softmax activation function which gives a probability distribution over the k classes is added on top of the hidden layer for modeling multi-class probabilities. The softmax function converts the classification result into label probabilities, i.e. $y' \in [0, 1]^k$.

The final training objective is to minimize the multinomial cross-entropy loss of the predicted and true distributions, where the error over a batch of n documents is calculated as:

$$\xi = -\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{k} y_{ij} \log \left(y'_{ij} \right) \tag{2}$$

where i denotes the ith training sample, j denotes the jth class, y is the true distribution (one-hot representation of the emotion label), and y' is the predicted probability distribution, $y'_{ij} \in [0, 1]$ and $\sum_j y'_{ij} = 1$.

3.1.2 Model 2: $\text{EWE}_{\text{MULTI}}$

Although modeling emotion classification as a multi-class problem captures the basic emotion connotation of many words, in reality, most words can be associated with more than one emotion. For instance, during the process of creating the NRC EmoLex emotion lexicon (Mohammad and Turney, 2013), it was found that *anger* words tend to be associated with *disgust*, *joy* terms tend to be related with *trust*, and *surprise* terms are largely also associated with *joy*.

In order to capture a word's association with more than one emotion, the $\text{EWE}_{\text{MULTI}}$ models multi-label classification setup where each document can belong to multiple emotion classes at the same time.

Assuming k emotion classes, and more than one valid emotion label for each document, the target variable y is binary represented. In other words, $y_j = 1$ indicates presence of an emotion class, and $y_j = 0$ otherwise. For example, if document d is labeled with a subset of emotion classes, $s_i \subseteq \mathcal{L}$, then:

$$y_j = \begin{cases} 1, & \text{if } y_j \in s_i \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

To predict the emotion label of the input text, an output layer with a sigmoid activation function, which squashes the inputs into a probability range of $[0, 1]$ for every class, is added to the last layer for modeling the probability of each class independently from the other classes.

The loss objective in this case is binomial cross-entropy, computed as follows:

$$\xi = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log\left(y_i'\right) + (1 - y_i)\log\left(1 - y_i'\right)\right] \tag{4}$$

where i denotes the ith training sample, y is the binary representation of true emotion label, and y' is the predicted probability.

3.1.3 Implementation

We use pre-trained word embeddings (GloVe $|\mathcal{V}| = 1.9\text{M}$, $d = 300$ (Pennington et al., 2014)) to initialize E and use random initialization sampled from a zero mean Gaussian distribution: $x \sim \mathcal{N}\left(0, \sigma^2\right)$ for words not found in the pre-trained embeddings. Optimization of the loss function is carried out with the Adam optimizer (Kingma and Ba, 2014), which is known for yielding quicker convergence, with learning rate of 0.001, and mini-batch size set to 1024.

3.2 Labeling Training Data using Distant Supervision

To learn the emotion embeddings, we require a large dataset of text with corresponding emotion labels. Due to the challenges involved in creating large-scale emotion resources (Mohammad and Turney, 2013), however, most existing manually-annotated emotion datasets contain a very limited number of instances and words. For example, two popular emotion datasets created by Alm (2008) and Aman and Szpakowicz (2007) contain around 1200 sentences each and only about 5000 unique words each. At the same time, in order for learned word representation models to be useful, they need to generalize well to diverse domains and applications by including a much larger number of words. For instance, the vocabulary size of most existing word representations is orders of magnitude larger (e.g., 400K to 1.9M words in GloVe (Pennington et al., 2014), 3M words in word2vec (Mikolov et al., 2013a), and so on).

As it is quite challenging to create a large manually annotated emotion dataset due to human time and effort required, we leverage distant supervision (Go et al., 2009) to create a weakly labeled training dataset automatically in order to obtain emotion-enriched word representations for a much larger vocabulary. Distant supervision is the process of labeling instances based on some heuristics or rules, with some of the instances being assigned noisy or imprecise labels.

3.2.1 Distant Supervision for EWE_{UNI}

Let $\mathcal{D} = \{d_1, d_2, ..., d_D\}$ be the set of unlabeled documents. The goal is to generate an annotated dataset $\mathcal{D} = \{(d_1, l_1), ..., (d_D, l_D)\}$, where $l_i \in \mathcal{L}$ is the corresponding emotion label for document d_i and $\mathcal{L} = \{l_1, ..., l_k\}$ is a known finite set of emotion labels.

Let $d = \left\{w_1, w_2, ..., w_{|d|}\right\}$ denote the sequence of words in a document, $w_i \in d$. For each word w_i, we compute an emotion vector $\mathbf{a}(w) = \langle a_1, a_2, ..., a_k\rangle$, where a_j indicates the word-emotion association as derived from a lexicon. Although technically, while any emotion taxonomy can be followed for deriving the word-emotion vector $\mathbf{a}(w)$, in this work, we adopt Ekman's (1992) model of six emotions (*anger*, *disgust*, *fear*, *happiness*, *sadness* and *surprise*), whose origins are firmly grounded and extensively verified in psychology. To this end, we select **WordNetAffect (WNA)** (Strapparava and Valitutti, 2004), which was developed by manually labeling the emotions of a few seed words and extending it to all their WordNet synonyms, and **NRC EmoLex (NRC)** (Mohammad and Turney, 2010; Mohammad

and Turney, 2013), which was created through crowdsourcing by annotating unigrams with one or more of Plutchik's (1980) eight emotions, which in turn is a superset of Ekman's (1992) model. In WNA, each word is associated with only one emotion, therefore $a_j = 1$ if w is associated with that emotion, and $a_j = 0$ otherwise. On the other hand, in NRC, a word can be binary associated with more than one emotion, with 1 indicating an association and 0 denoting no association. For a given word w, we extract its binary association scores corresponding to the six categories of Ekman's model.

The emotion vector $\mathbf{a}(d)$ for document d is then, the sum of the emotion vectors of all its words, $\mathbf{a}(d) = \sum_{i \in d} \mathbf{a}(w_i)$. If the document has an association with at least one emotion, i.e., $\exists j, a_j(d) > 0$, then, $S = \text{argmax}_i\ \mathbf{a}(d)$, where $S \subseteq \mathcal{L}$. In other words, documents assigned zero emotion score are not considered. Finally, in case multiple emotion labels have the maximum value, i.e., $|S| > 1$, we sample uniformly at random one emotion label $l \in S$.

We investigate two strategies of computing the affective knowledge: (i) **one lexicon** - where any one lexicon is used to guide the labeling process; and, (ii) **two or more lexicons** - whereby two or more lexicons are used in order to mitigate some effects of noisy labeling. This variant assigns an emotion label to a document only if the labels output by both the lexicons match.

3.2.2 Distant Supervision for $\text{EWE}_{\text{MULTI}}$

Some words evoke more than one emotion at the same time. For example, out of the 14,000 words annotated with emotions in the NRC lexicon, almost 8,000 words (57%) are associated with more than one emotion. Therefore, we relax the labeling scheme followed in EWE_{UNI} and design $\text{EWE}_{\text{MULTI}}$ to take into consideration a multi-class, multi-label setting, where a document can have more than one emotion label.

Unlike EWE_{UNI}, in $\text{EWE}_{\text{MULTI}}$ the set of all emotions with $a_j(d) > 0$ for document d is used as final emotion labels for d. Thus, the multi-label annotated dataset $\mathcal{D}$ is $\{(d_1, S_1), ..., (d_n, S_n)\}$, where each document d_i is assigned a set of emotion labels, $S_i \subseteq \mathcal{L}$

3.2.3 Training Data

Our large corpus of unlabeled documents is extracted from the Amazon reviews dataset (McAuley et al., 2015) consisting of product reviews, spanning May 1996 - July 2014. Each review (considered as a document) is preprocessed by converting it to lowercase, tokenizing it with the NLTK toolkit (punctuation is preserved as tokens), and filtering out reviews that are too short (less than 5 words). Note that, as the proposed weak labeling is not dependent on any domain-specific indicators of affect such as emoticons or hashtags, it can be easily generalized to any type of text documents.

4 Experiments

4.1 Emotion Classification

The first task validates the effectiveness of the emotion embeddings under the supervised framework of emotion classification, where the learned word vectors are fed as features into classification models for predicting the emotion labels. We train two classifiers: (i) L2-regularized multi-class logistic regression (LR) and (ii) support vector machine (SVM) based on LIBSVM (Chang and Lin, 2011), to predict the fine-grained emotion label at the sentence level. The results of 10-fold cross validation are reported in terms of macro F_1 score, which is the average F_1 score over all the emotion classes. F_1 score is the harmonic mean of precision (p) and recall (r), $F_1 = 2\frac{p \cdot r}{p+r}$.

For the emotion lexicons, we generate a feature vector consisting of the total number of words in the sentence associated with each emotion category. For the word embedding models, we compute the average of the word vectors of all the words in the sentence along each dimension to obtain the sentence representation as the input to the classification algorithm.

4.1.1 Emotion Datasets

The following four benchmark emotion datasets from various genres of text are considered for emotion classification. The statistics of the datasets are summarized in Table 2.

dataset	domain	# emotions	total
Alm	fairy tales	5	1207
Aman	blogs	6	1290
ISEAR	experiences	5	5412
EmoTweet-top8	tweets	8	4664

Table 2: Statistics of emotion datasets

Methods		Alm	Aman	ISEAR
Lexicons	WNA	0.459	0.405	0.384
	NRC	0.387	0.370	0.378
	WNA+NRC	0.521	0.474	0.465
EWE_{UNI}	WNA	0.635	0.604	0.674
	NRC	0.604	0.582	0.666
	WNA+NRC	**0.661**	**0.623**	**0.679**
EWE_{MULTI}	NRC	0.630	0.602	0.666

Table 3: Comparison of using lexicons directly versus using lexicons to guide representation learning.

Alm: Emotions are particularly significant in the literary genre of fairy tales and this dataset contains sentences marked with one of five emotion categories: *angry-disgusted, fearful, happy, sad* and *surprised* (Alm, 2008).
Aman: Consisting of highly informal blog data, this dataset includes sentences annotated with one of six emotions: *anger, disgust, fear, joy, sadness* and *surprise* (Aman and Szpakowicz, 2007).
ISEAR: Developed for studying the relationships among cultures and emotions, this dataset contains personal experiences evoking seven emotions (Wallbott and Scherer, 1986). We use a subset of this dataset marked with one of the five emotions: *anger*, *disgust*, *fear*, *joy* and *sadness*.
EmoTweet-28: While the other annotated datasets are modeled after existing emotion taxonomies, this corpus was created by inductively identifying a set of emotion categories that characterize the emotions expressed in tweets (Liew et al., 2016). For our experiments, we extract a subset (**EmoTweet-top8**) of the eight most frequent emotions in the dataset.

4.1.2 Lexicons versus Representations

As the quality of the emotion embeddings depends on the underlying emotion lexicons adopted to create the training data, we analyze the results of using the source emotion lexicons directly versus using them to initialize EWE in Table 3.

We observe that the configurations using both the lexicons (WNA+NRC) yield better results than using any one lexicon alone. Morevover, all the EWE embeddings demonstrate significant improvements over using the lexicons directly, indicating that the affective word representation model learns useful information in addition to the knowledge available in the base lexicons adopted during distant supervision to guide the representation learning process.

4.1.3 Comparison Against State-of-the-art Representations

Next, we analyze the performance of EWE against state-of-the-art *generic embeddings* and *task-specific affective embeddings* described below, and summarized in Table 4.
Generic Embeddings: **(i) GloVe**: Global vectors[2] for word representations (Pennington et al., 2014) trained on aggregated global word-word co-occurrence statistics from a corpus capture linear substructures of the word vector space. We use the vectors that were trained on: **GloVe 6B**: 6 billion words, uncased, from Wikipedia 2014 and Gigaword v5, of dimension $d = 300$; **GloVe 42B**: 42 billion words,

[2]http://www-nlp.stanford.edu/projects/ glove/

embeddings	corpus	size	$\|\mathcal{V}\|$
GloVe 6B	Wiki + Gigaword	6B tokens	400K
GloVe 42B	Common Crawl	42B tokens	1.9M
word2vec	Google news	100B tokens	3M
SSWE	Twitter tweets	10M tweets	137K
DeepMoji	Twitter tweets	1B tweets	50K
EWE	Amazon reviews	200K reviews	183K

Table 4: Details of compared embeddings

methods		Alm		Aman		ISEAR		EmoTweet-top8	
		LR	SVM	LR	SVM	LR	SVM	LR	SVM
GloVe 6B	d =300	0.548	0.583	0.547	0.555	0.648	0.643	0.574	0.581
GloVe 42B	d =300	<u>0.590</u>	<u>0.624</u>	<u>0.564</u>	<u>0.609</u>	<u>0.675</u>	<u>0.671</u>	<u>0.609</u>	<u>0.614</u>
word2vec	CBOW	0.413	0.433	0.424	0.478	0.655	0.661	0.526	0.568
SSWE	u	0.368	0.371	0.363	0.363	0.495	0.505	0.443	0.444
DeepMoji	d =256	0.300	0.275	0.332	0.336	0.598	0.607	0.533	0.560
Retrofitting	GloVe 42B	0.141	0.110	0.111	0.111	0.553	0.559	0.245	0.220
Retrofitting	word2vec	0.110	0.108	0.100	0.098	0.488	0.472	0.232	0.214
EWE_{UNI}	WNA+NRC	**0.632**	**0.661**	**0.602**	**0.623**	**0.679**	**0.679**	**0.610**	**0.618**

Table 5: Comparison against state-of-the-art word representations (*generic embeddings* in the top half; *affective embeddings* in the bottom half) on emotion classification. The best results are shown in **bold**, and the second best results are <u>underlined</u>. Paired t-tests using the results on all four datasets indicate EWE is significantly better than all the other methods with p-values < 0.02.

uncased, from Common Crawl, of dimension $d = 300$. **(ii) `word2vec`**: These word representations[3] were learned with a continuous bag-of-words model (CBOW) (Mikolov et al., 2013a), where a target word is predicted given its surrounding context words. We use the vectors that were trained on 100 billion words of Google news dataset and are of $d = 300$.

Affective Embeddings: **(i) Sentiment-specific word embeddings (SSWE)**: These embeddings, obtained using a corpus of 10 million tweets, encode the sentiment information (derived using a set of positive and negative emoticons) of the text in the continuous representation of words[4] (Tang et al., 2014). We use embeddings that were trained with the unified model (SSWE_u). **(ii) DeepMoji**: These word representations were obtained from a corpus of almost 1 billion tweets weakly labeled using a set of 64 emojis (Felbo et al., 2017). **(iii) Retrofitting**: Instead of directly training task-specific affective embeddings such as SSWE and DeepMoji, Retrofitting (Faruqui et al., 2015) is a post-processing technique of tuning existing embeddings according to a task-specific lexicon. Using WNA as the source emotion lexicon, where words with same emotions are clustered together, we apply Retrofitting to the generic word vectors (GloVe and word2vec).

The results of the emotion classification are presented in Table 5, with the generic embeddings model in the top half and affective embeddings in the bottom half. In general, we observe that GloVe 42B yields the second best results overall, and in line with other recent studies (Pool and Nissim, 2016), Retroffiting did not improve over any original word embeddings suggesting that post-processing word embeddings with respect to emotion knowledge requires additional considerations.

Secondly, although SSWE and DeepMoji were both trained on tweets data, they perform very differently to each other, most likely due to their extremely different choices of affect spectrum (SSWE was

[3]https://code.google.com/p/word2vec

[4]http://ir.hit.edu.cn/~dytang/paper/sswe/embedding-results.zip

embedding	$n = 10$	$n = 20$	$n = 30$
SSWE_u	32.6	28.8	28.2
word2vec	35.5	33.1	30.2
GloVe	35.1	32.5	30.4
$\text{EWE}_{\text{UNI(WNA+NRC)}}$	**36.7**	**33.2**	**31.3**

Table 6: Accuracy of emotion similarity tested on emotion lexicon DepecheMood

modeled along binary polarities, whereas DeepMoji used an axis of 64 categories), thus highlighting the importance of the emotion model adopted for creating the training dataset. In addition, and rather surprisingly, all the generic embeddings (GloVe and word2vec) outperform all the affective embeddings (SSWE and DeepMoji) on all the four datasets. One possible reason for this could be due to the more generalizable sources of data that were used to induce the generic embeddings, while the affective embeddings were trained on tweets data, thus showing the significance of the choice of the underlying text used to derive the representations.

Lastly, $\text{EWE}_{\text{UNI(WNA+NRC)}}$ statistically significantly outperforms all the other baselines across all the four datasets, indicating the effectiveness of the proposed method.

4.2 Emotion Similarity

The second task measures the *emotion similarity* of the word vectors by comparing against the emotion similarity obtained from an emotion lexicon. In this experiment, the test affective information is derived from **DepecheMood (DM)** (Staiano and Guerini, 2014), an emotion lexicon consisting of 37,000 words and their emotion scores across eight affective dimensions. We consider the emotion label of a word as the emotion category with the maximum affective weight.

Following previous experimental setup for measuring affective consistency (Tang et al., 2014), we compute the accuracy of emotion similarity consistency between each emotion word and its top n nearest neighboring words as follows:

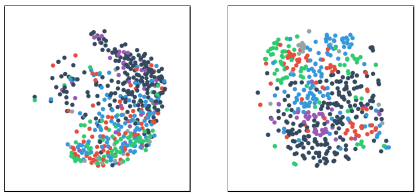
$$Accuracy = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\alpha\left(w_i, c_{ij}\right)}{m \times n} \tag{5}$$

where m is the number of words in the emotion lexicon, w_i is the ith word in the lexicon, c_{ij} is the jth closest word to w_i in terms of their cosine similarity, $\alpha\left(w_i, c_{ij}\right)$ is an indicator function, where $\alpha = 1$ if w_i and c_{ij} belong to the same emotion category and $\alpha = 0$ otherwise. The higher the accuracy, the better the clustering of emotionally similar words in the embedding space.

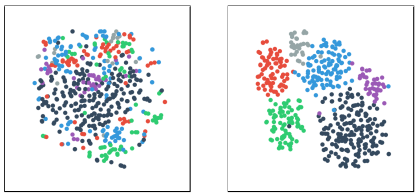
Table 6 presents the results of various embeddings for $n = \{10, 20, 30\}$, where n is the number of nearest neighboring words. For fair comparison, for each word embeddings, only the words that appear in both the vocabularies (i.e., DM and word embeddings) have been used. Again, we observe that generic embeddings such as GloVe and word2vec outperform affective embeddings such as SSWE. The best results are obtained from EWE which have been specifically trained to capture emotion similarity.

5 Qualitative and Error Analysis

To further analyze the learned emotion embedding space, we use t-SNE (van der Maaten and Hinton, 2008) to visualize the word representations of a small subset of words in Figure 2. The plots show that compared to other models, EWE is effective in clustering emotionally similar words into neighboring vector spaces. Figure 3 shows confusion matrix plots providing an overview for some error analysis. In general, for imbalanced datasets such as Alm and Aman, it is observed that most misclassified instances are incorrectly labeled as *happy* class, likely because the *happy* class contains a disproportionately large number of training instances. Moreover, instances belonging to the *surprise* class are more often misclassified than correctly predicted, likely because the *surprise* class is highly underrepresented. Balancing

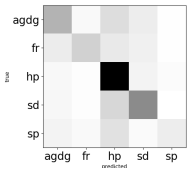


(i) SSWE (ii) word2vec

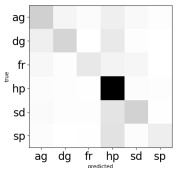


(iii) GloVe (iv) EWE

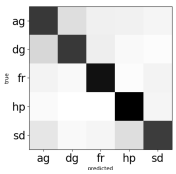
Figure 2: t-SNE visualization of word embeddings



(i) Alm



(ii) Aman



(iii) ISEAR

Figure 3: Confusion matrix error analysis

the datasets might prove helpful. In ISEAR, *anger* and *disgust* classes are found to be confused with each other and *sadness* seems to be challenging.

6 Conclusions

In this paper, we described a novel method of learning emotion-enriched word representations by leveraging distant supervision and neural networks. Significant improvements over baseline representations in two tasks including emotion classification and emotion similarity is obtained. In addition, we presented a qualitative analysis of the learned word vectors. As future work, we plan on considering alternate taxonomies of emotions such as Plutchik's (1980), obtaining emotion-enriched representations at phrase level and exploring sentence compositionality.

Acknowledgments

This work is funded by Natural Sciences and Engineering Research Council of Canada (NSERC) and Big Data Research, Analytics, and Information Network (BRAIN) alliance established by the Ontario Research Fund - Research Excellence Program (ORF-RE).

References

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: Machine learning for text-based emotion prediction. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 579–586, Stroudsburg, PA, USA. ACL.

Ebba Cecilia Ovesdotter Alm. 2008. *Affect in Text and Speech*. Ph.D. thesis, University of Illinois at Urbana-Champaign.

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. In *Proceedings of the 10th International Conference on Text, Speech and Dialogue*.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 809–815.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Felipe Bravo-Marquez, Eibe Frank, Saif M Mohammad, and Bernhard Pfahringer. 2016. Determining word-emotion associations from tweets by multi-label classification. In *Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on*, pages 536–539. IEEE.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *EMNLP*. ACL.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 160–167, New York, NY, USA. ACM.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, nov.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & Emotion*, 6(3-4).

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard H. Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1606–1615.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *EMNLP*.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *Processing*, pages 1–6.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780, November.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 655–665, Baltimore, Maryland, June. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Igor Labutov and Hod Lipson. 2013. Re-embedding words. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 2: Short Papers*, pages 489–493.

Jasy Suet Yan Liew, Howard R. Turtle, and Elizabeth D. Liddy. 2016. Emotweet-28: A fine-grained emotion corpus for sentiment analysis. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.*

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-based recommendations on styles and substitutes. In *SIGIR*, pages 43–52. ACM.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

Andriy Mnih and Geoffrey E. Hinton. 2008. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008*, pages 1081–1088.

Saif M. Mohammad and Felipe Bravo-Marquez. 2017. Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*, Vancouver, Canada.

Saif M. Mohammad and Peter D. Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 26–34, Stroudsburg, PA, USA. Association for Computational Linguistics.

Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a word-emotion association lexicon. 29(3):436–465.

W. Parrott, Gerrord. 2001. *Emotions in Social Psychology*. Psychology Press, Philadelphia.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Robert Plutchik, 1980. *A general psychoevolutionary theory of emotion*, pages 3–33. Academic press, New York.

Chris Pool and Malvina Nissim. 2016. Distant supervision for emotion detection using facebook reactions. *arXiv preprint arXiv:1611.02988*.

Lizhen Qu, Gabriela Ferraro, Liyuan Zhou, Weiwei Hou, Nathan Schneider, and Timothy Baldwin. 2015. Big data small data, in domain out-of domain, known word unknown word: The impact of word representations on sequence labelling tasks. In *Proceedings of the 19th Conference on Computational Natural Language Learning, CoNLL 2015, Beijing, China, July 30-31, 2015*, pages 83–93.

Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 151–161.

Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. 2013. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 455–465.

Jacopo Staiano and Marco Guerini. 2014. Depechemood: a lexicon for emotion analysis from crowd-annotated news. *CoRR*, abs/1405.1605.

Carlo Strapparava and Alessandro Valitutti. 2004. WordNet-Affect: An affective extension of WordNet. In *LREC*, pages 1083–1086.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *ACL*.

Duyu Tang, Furu Wei, Bing Qin, Nan Yang, Ting Liu, and Ming Zhou. 2016. Sentiment embeddings with applications to sentiment analysis. *IEEE Trans. Knowl. Data Eng.*, 28(2):496–509.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 384–394, Stroudsburg, PA, USA. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey E. Hinton. 2008. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Harald G. Wallbott and Klaus R. Scherer. 1986. How universal and specific is emotional experience? evidence from 27 countries on five continents. *Social Science Information*, 25(4):763–795.