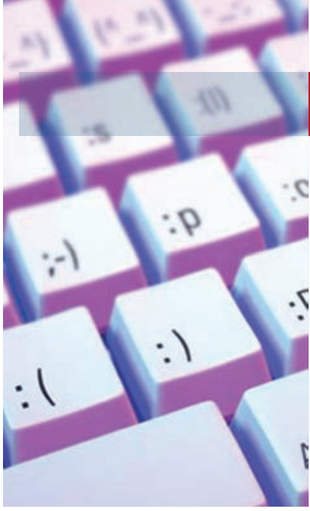

Editor: **Erik Cambria,** Nanyang Technological University, Singapore, cambria@ntu.edu.sg

Lexicon Generation for Emotion Detection from Text

Anil Bandhakavi, Nirmalie Wiratunga, and Stewart Massie, *Robert Gordon University*
Deepak Padmanabhan, *Queen's University, Belfast*

Textual emotion detection is the computational study of natural language expressed in text in order to identify its association with emotions such as anger, fear, joy, and sadness. It has potential in many different applications for industry, media, and government. However, its uptake arguably has been slow, mainly because of the challenges involved in modeling fine-grained subjectivity and the subtlety of emotive expressions in text.

Until recently, popular resources such as sentiment lexicons[1] and general-purpose emotion lexicons (GPELs) such as WordNet-Affect[2] have been used for emotion detection from text (see the "Related Work in Lexicon Generation" sidebar for more information). However, both sentiment lexicons and GPELs are inadequate for emotion detection in inherently dynamic domains (such as social media) because the former lack granular emotion information and the latter have a static and formal nature. For instance, on Twitter, informal vocabulary and emoticons are used to convey emotions, instead of formal vocabulary as in GPELs. Furthermore, the association between words and emotions varies from one domain to another and calls for contextual disambiguation. For example, "glee" might normally indicate joy, but it would need to be interpreted as neutral in a corpus of documents talking about the television series with the same name. Furthermore, "unfair" might be associated with anger despite being more dominant in documents expressing sadness; the crisp binary memberships of words in GPELs cannot capture such fuzzy associations between words and emotion classes. Therefore, it is necessary to build domain-specific emotion lexicons (DSELs) that offer quantitative fine-grained estimates for word-emotion associations within a domain. Accordingly, recent efforts in emotion detection focused on learning emotion lexicons from labeled emotion corpora as well as weakly labeled social media content.[3–6]

Social media offers access to users' weakly labeled emotional data containing emoticons and emotion hashtags, which can be leveraged to learn DSELs for various emotion-detection tasks. In particular, DSELs offer useful knowledge to design a range of document representations from simple binary to frequency counts to sophisticated emotion concepts. Furthermore, DSELs can be deployed to search and index vast amounts of emotional content (such as song lyrics and video descriptions) on the social web in order to infer emotions of social groups and communities.

Our contributions in this article are threefold. First, we propose a generative unigram mixture model (UMM) to learn a word-emotion association lexicon from an input document corpus. Second, we empirically evaluate the quality of the emotion language models (topics) generated by the proposed method and supervised latent Dirichlet allocation (sLDA) using standard metrics, such as perplexity. Finally, we evaluate the quality of the emotion lexicons generated by the proposed method and state-of-the-art baseline methods on two emotion-detection tasks: word-emotion classification and document-emotion ranking.

Problem Definition

The problem essentially is to learn a word-emotion lexicon from an input corpus of emotion-labeled documents. Given a corpus of documents X, with emotion labels from $E = \{e_1, \ldots, e_k\}$, we learn a word-emotion lexicon Lex, in which Lex(i, j) is the

1541-1672/17/$33.00 © 2017 IEEE
Published by the IEEE Computer Society

Related Work in Lexicon Generation

Emotion lexicons, unlike sentiment lexicons, offer granular emotion information.[1,2] WordNet synsets were manually labeled with Paul Ekman's basic emotions[3] to generate WordNet-Affect.[4] The NRC word-emotion lexicon[5] was obtained by crowdsourcing emotion annotations to 14,182 words from the Google Ngram corpus (see http://catalog.ldc.upenn.edu/LDC2006T13). As opposed to earlier lexicons, researchers have proposed semantically rich lexicons such as SenticNet[6,7] to model the sentiment of multiword expressions using commonsense knowledge derived from ConceptNet.[8] Further fuzzy clustering and machine learning techniques are applied to assign WordNet-Affect emotion labels to concepts in SenticNet to obtain EmoSenticNet.[9] A common limitation of the aforementioned emotion lexicons is that their vocabulary is static and formal, which makes it challenging to deploy them in dynamic and informal domains (such as social media) for emotion detection. To address this limitation, researchers have proposed methods for building lexicons that capture the domain-level associations between words and emotions.[10–12]

Existing methods for building domain-specific lexicons are mostly supervised, because they rely on either labeled or weakly labeled emotive content in a domain. For instance, researchers applied Pointwise Mutual Information to learn a word-emotion lexicon from tweets weakly labeled with emotion hashtags.[13] Jacopo Staiano and Marco Guerini proposed leveraging crowd-annotated emotional news articles (www.rappler.com) for lexicon generation by combining the document-frequency distributions of words and the emotion distributions over documents.[10]

In addition, researchers have applied generative models such as latent Dirichlet allocation (LDA) to lexicon generation. Yanghui Rao and colleagues combined user emotion ratings on documents (http://news.sina.com.cn/society [in Chinese]), document-frequency distributions, and document-topic distributions from LDA to learn word-emotion, topic-emotion lexicons.[14] Min Yang and colleagues proposed a semisupervised LDA approach, which uses a minimal set of domain-independent emotion seed words to guide the LDA process to learn emotion-relevant topics.[15] However, the topics learned from this approach are not consistently accurate, because the coverage of seed words varies from one domain to another. Nevertheless, supervised LDA (sLDA)[16] offers a more accurate means to learn emotion-topic models for lexicon generation from labeled or weakly labeled emotion corpora.

In this article, we propose a mixture model for learning a domain-specific word-emotion lexicon. Our model assumes documents to be a mixture of emotional and neutral words, which is different from the generative model of sLDA that assumes documents are a mixture of multiple emotion (topic) words. We expect the joint modeling of emotionality and neutrality at the word level to be more effective on real-world emotion corpora, because not every word in them connotes emotions.

References

1. A. Esuli, S. Baccianella, and F. Sebastiani, "Sentiwordnet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining," *Proc. Int'l Conf. Language Resources and Evaluation*, 2010, pp. 18–38.
2. Y. Chen and S. Skiena, "Building Sentiment Lexicon for All Major Languages," *Proc. 52nd Ann. Conf. Assoc. Computational Linguistics*, 2014, pp. 383–389.
3. P. Ekman, "An Argument for Basic Emotions," *Cognition and Emotion*, vol. 6, no. 3, 1992, pp. 169–200.
4. C. Strapparava and A. Valitutti, "WordNet-Affect: An Affective Extension of WordNet," *Proc. Int'l Conf. Language Resources and Evaluation*, 2004, pp. 1083–1086.
5. S.M. Mohammad and P. Turney, "Crowdsourcing a Word-Emotion Association Lexicon," *Computational Intelligence*, vol. 29, no. 3, 2013, pp. 436–465.
6. E. Cambria et al., "SenticNet 4: A Semantic Resource for Sentiment Analysis Based on Conceptual Primitives," to be published in *Proc. 26th Int'l Conf. Computational Linguistics*, 2016.
7. E. Cambria, "Affective Computing and Sentiment Analysis," *IEEE Intelligent Systems*, vol. 31, no. 2, 2016, pp. 102–107.
8. H. Liu and P. Singh, "ConceptNet: A Practical Commonsense Reasoning Tool-kit," *BT Technology J.*, vol. 22, no. 4, 2004, pp. 211–226.
9. S. Poria et al., "EmoSenticSpace: A Novel Framework for Affective Common-Sense Reasoning," *Knowledge-Based Systems*, vol. 69, 2014, pp. 108–123.
10. J. Staiano and M. Guerini, "DepecheMood: A Lexicon for Emotion Analysis from Crowd-Annotated News," *Proc. 52nd Ann. Conf. Assoc. Computational Linguistics*, 2014, pp. 427–433.
11. K. Song et al., "Build Emotion Lexicon from Microblogs by Combining Effects of Seed Words and Emoticons in a Hetereogeneous Graph," *Proc. 26th ACM Conf. Hypertext & Social Media*, 2015, pp. 283–292.
12. S. Bao et al., "Mining Social Emotions from Affective Text," *IEEE Trans. Knowledge and Data Eng.*, vol. 24, no. 9, 2012, pp. 1658–1670.
13. S.M. Mohammad, "#emotional tweets," *Proc. 1st Jt. Conf. Lexical and Computational Semantics*, 2012, pp. 246–255.
14. Y. Rao et al., "Building Emotional Dictionary for Sentiment Analysis of Online News," *World Wide Web*, vol. 17, no. 4, 2014, pp. 723–742.
15. M. Yang et al., "A Topic Model for Building Fine-Grained Domain-Specific Emotion Lexicon," *Proc. 52nd Ann. Conf. Assoc. Computational Linguistics*, 2014, pp. 421–426.
16. D.M. Blei and J.D. McAuliffe, "Supervised Topic Models," *Advances in Neural Information Processing Systems*, 2007, pp. 121–128.

emotional valence of the ith word in vocabulary V to the jth emotion in E, and Lex(i, $k + 1$) corresponds to its neutral valence. The word-emotion lexicon is using a set of k UMMs, where the tth UMM assumes that documents in X labeled with emotion e_t are a mixture of words bearing e_t and some background (neutral) words. Therefore, each UMM is a linear combination of two unigram language models, θ and N, along with a mixing parameter λ.

Figure 1 shows the conceptual diagram of the proposed UMM. Initial models $\theta_{e_t}^{(0)}$ and N are learned from the training data. Mixture parameter λ_{e_t} is set empirically. The estimation of the hidden variable, Z_w, happens in the

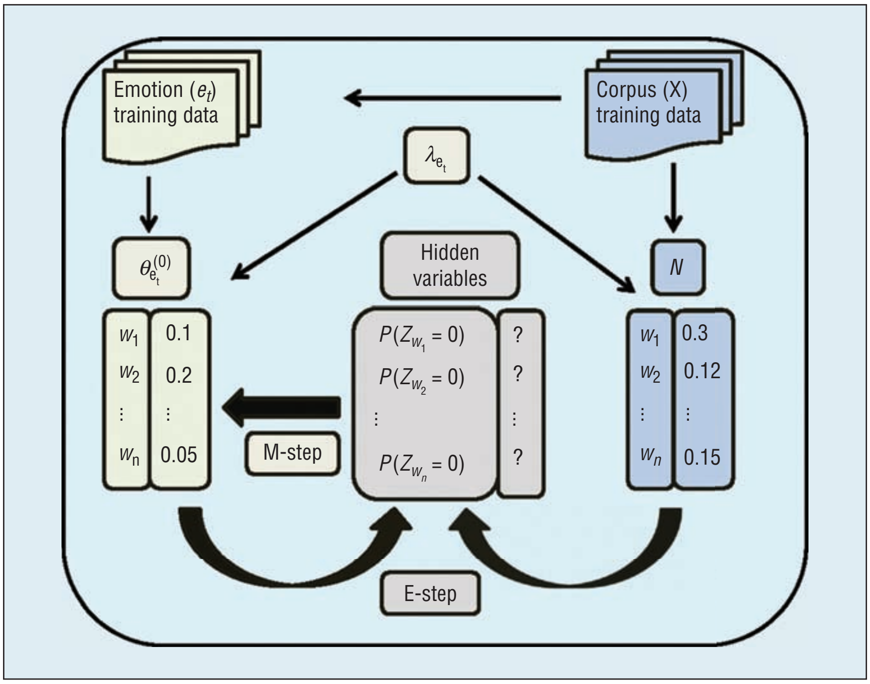

Figure 1. Visualization of the unigram mixture model (UMM) generation and the expectation/maximization (EM) iterative process for emotion e_t.

Table 1. Important notations.

Notation	Description
X	Corpus of emotion-labeled documents
E	Set of emotion labels
D_{e_t}	Documents labeled with emotion e_t
N	Neutral (background) language model
θ_{e_t}	Language model for e_t
V	Set of unique words from documents in X
w_i	ith word in vocabulary V
Z_{w_i}	Hidden (unobserved) variable corresponding to w_i
λ_{e_t}	Mixture parameter (empirically estimated)
n	EM iteration number
$Q\left(\theta_{e_t}^{(n+1)};\theta_{e_t}^{(n)}\right)$	Q-function
$c(w, d_i)$	No. times word w occurs in document d_i
Lex(i, j)	Emotional valence between word w_i and emotion e_j
Lex$(i, k+1)$	Neutral valence for the word w_i

expectation step (E-step). In the maximization step (M-step) parameter, θ_{e_t} is updated. This process repeats until the values of θ_{e_t} do not change significantly. Table 1 summarizes the important mathematical notations.

Generative Model for Documents

We outline our generative model for emotion-bearing documents using an example from real-world data. Real-world emotion data typically is a mixture of emotion-rich words and background (emotion-neutral) words. For example, consider the tweet "Sunday in LasVegas #excited #joyous," which explicitly connotes the emotion *joy*. The word "Sunday" does not evidently express joy. Furthermore, Las Vegas could connote other emotions, such as love. Therefore, it is important to have a model that accounts for such word mixtures in the documents. The mixture model in our case is as follows. Let D_{e_t} be the documents labeled with e_t; then, according to the UMM, documents in D_{e_t} are generated independently from a linear mixture of an emotion language model θ_{e_t} and a background language model N as shown in Equation 1:

$$P\left(D_{e_t}, Z \middle| \theta_{e_t}\right) = \prod_{i=1}^{|D_{e_t}|} \prod_{w \in d_i} \left[(1-Z_w)\lambda_{e_t} P\left(w \middle| \theta_{e_t}\right) + (Z_w)\left(1-\lambda_{e_t}\right) P\left(w \middle| N\right)\right]^{c(w,d_i)}. \quad (1)$$

Note that this mixture model reduces to a single language model when λ_{e_t} is 1. Thus, λ_{e_t} in our case indicates the noisy (neutral and other emotion) words that occur in documents connoting e_t. Finally, Z_w is the hidden (latent) binary variable corresponding to word w, which indicates the mixture component (language model) that generated w. For each word $w \in V$, its corresponding hidden variable is defined as

$$Z_w = \begin{cases} 1 & \text{if word } w \text{ is from the neutral model} \\ 0 & \text{otherwise.} \end{cases}$$

In the rest of this article, we will illustrate the estimation of parameters (θ_{e_t}, λ_{e_t}, and Z) of the mixture model, followed by lexicon generation.

Parameter Estimation of the Mixture Model

The objective is to find the parameters (θ_{e_t}, λ_{e_t}, and Z) that maximize the probability of generating documentsD_{e_t}. We can estimate λ_{e_t} using maximum likelihood estimation (MLE) as shown in Equation 2:

$$\hat{\lambda}_{e_t} = \arg\max_{\lambda_{e_t}} \sum_{i=1}^{|D_{e_t}|} \sum_{w \in d_i} c(w, d_i) \times \log[\lambda_{e_t} P(w \mid \theta_{e_t}) + (1-\lambda_{e_t}) P(w|N)]. \quad (2)$$

The estimation of parameters θ_{e_t} and Z can be done using expectation maximization (EM), which iteratively maximizes the complete data (D_{e_t}, Z) by alternating between the E-step and M-step. In the E-step, we estimate the value of the hidden variable (Z_w). Observe that

$$P\left(Z_w = 0 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) + P\left(Z_w = 1 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) = 1. \quad (3)$$

Further from Bayes' theorem, it follows that

$$P\left(Z_w = 0 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) = C \times \lambda_{e_t} \times P\left(w \middle| \theta_{e_t}^{(n)}\right). \quad (4)$$

Combining Equations 3 and 4 gives us Equation 5:

E-step:

$$P\left(Z_w = 0 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) = \frac{\lambda_{e_t} P\left(w \middle| \theta_{e_t}^{(n)}\right)}{\lambda_{e_t} P\left(w \middle| \theta_{e_t}^{(n)}\right) + \left(1-\lambda_{e_t}\right) P(w|N)}. \quad (5)$$

The M-step involves maximizing the function shown in Equation 6:

$$Q\left(\theta_{e_t}^{(n+1)}; \theta_{e_t}^{(n)}\right) = \sum_{i=1}^{|D_{e_t}|} \sum_{w \in d_i} c(w, d_i) \left[P\left(Z_w = 0 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) \log \left(\lambda_{e_t} P\left(w \middle| \theta_{e_t}^{(n+1)}\right)\right) + P\left(Z_w = 1 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) \log\left(\left(1-\lambda_{e_t}\right) P(w|N)\right)\right]. \quad (6)$$

We thus consider the auxiliary function, Equation 7,

$$g\left(\theta_{e_t}^{(n+1)}\right) = Q\left(\theta_{e_t}^{(n+1)}; \theta_{e_t}^{(n)}\right) + \mu\left(1 - \sum_{w \in V} P\left(w \middle| \theta_{e_t}^{(n+1)}\right)\right), \quad (7)$$

in which μ is the Lagrange multiplier. Computing the first-order partial derivative of $g\left(\theta_{e_t}^{(n+1)}\right)$s with respect to the parameter variable $P\left(w \middle| \theta_{e_t}^{(n+1)}\right)$ and equating to zero gives us Equation 8:

M-step:

$$p\left(w \middle| \theta_{\theta e_t}^{(n+1)}\right) = \frac{\sum_{i=1}^{|D_{e_t}|} P\left(Z_w = 0 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) c(w, d_i)}{\sum_{w \in V} \sum_{i=1}^{|D_{e_t}|} P\left(Z_w = 0 \middle| D_{e_t}, \theta_{e_t}^{(n)}\right) c(w, d_i)}. \quad (8)$$

Equation 9 defines the initial language model $\theta_{e_t}^{(0)}$ for e_t as

$$P\left(w_i \middle| \theta_{e_t}^{(0)}\right) = \frac{f\left(w_i, D_{e_t}\right)}{\sum_{w \in V} f\left(w_i, D_{e_t}\right)}, \quad (9)$$

in which $f\left(w_i, D_{e_t}\right)$ is the frequency of the ith word in V in the training documents for e_t. Equation 10 gives the background (neutral) language model as

$$P(w_i|N) = \frac{f(w_i, X)}{\sum_{w \in V} f(w_i, X)}, \quad (10)$$

in which $f(w_i, X)$ is the training corpus frequency for word w_i.

Lexicon Generation

The word-emotion lexicon is learned using k emotion language models and the background model N as shown in Equations 11 and 12:

$$\mathrm{Lex}^{(n)}\left(w_i, \theta_{e_j}\right) = \frac{P\left(w_i \middle| \theta_{e_j}^n\right)}{\sum_{t=1}^{k} \left[P\left(w_i \middle| \theta_{e_t}^n\right)\right] + P(w_i|N)} \quad (11)$$

$$\mathrm{Lex}^{(n)}\left(w_i, \theta_{e_j}\right) = \frac{P\left(w_i \middle| \theta_{e_j}^n\right)}{\sum_{t=1}^{k} \left[P\left(w_i \middle| \theta_{e_t}^n\right)\right] + P(w_i|N)}, \quad (12)$$

in which k is the number of emotions in the corpus and $\mathrm{Lex}^{(n)}$ is a $|V| \times (k+1)$ matrix generated after the nth EM iteration.

Lexicon Evaluation Tasks

In this section, we formulate the different evaluation tasks for assessing the lexicons' quality.

Word-Emotion Classification

In this task, we evaluate a lexicon's ability to classify a collection of target words hand-labeled with emotions. More formally, given an arbitrary word w, the task is to predict an emotion label $e \in E$ for w using the word-emotion lexicon. Because a DSEL quantifies the associations between words in a vocabulary V and a range of emotions in E, for any given arbitrary word w, the dominant emotion e being expressed is calculated using the lexicon as shown in Equation 13:

$$e = \arg\max_j \mathrm{Lex}(w, j). \quad (13)$$

In contrast, in a GPEL, Lex(w, j) is modeled as a list of words per class, as in Equation 14:

$$\mathrm{Lex}(w, j) = \begin{cases} 1 & \text{if } w \in \mathrm{List}\left(E_j\right) \\ 0 & \text{otherwise,} \end{cases} \quad (14)$$

in which List(E_j) is the word list for the jth emotion.

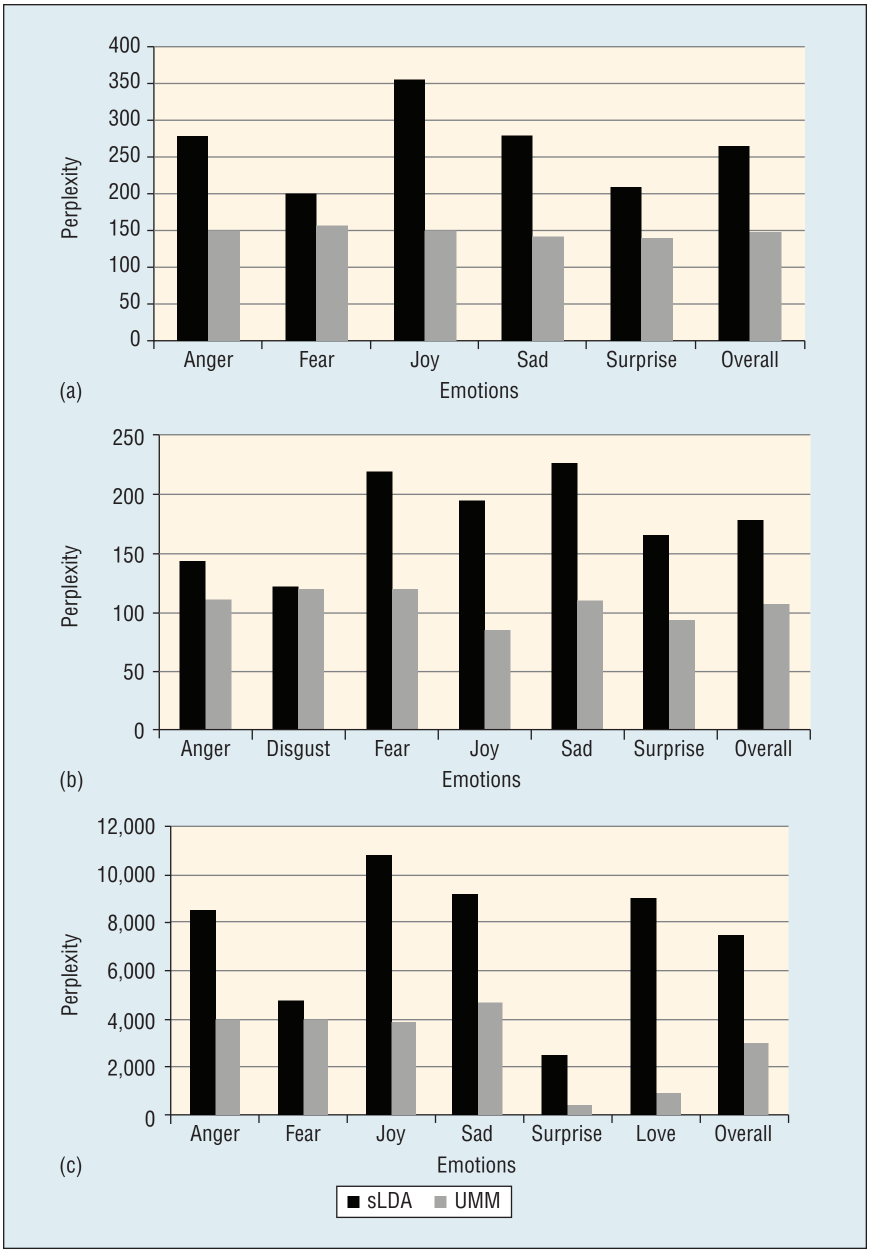

Figure 2. Results for perplexity analysis on (a) blogs, (b) news, and (c) tweets. UMM-generated emotion topics obtained significantly lower perplexity compared to sLDA-generated emotion topics.

Document-Emotion Ranking

In this task, we aim to assess the lexicon's quality in predicting the association between a document and multiple emotions. More formally, given a document d expressing emotions $(e_1, \ldots, e_m)$ in decreasing order of magnitude, the task is to predict the order of emotions for d using a lexicon. For any given document d, an emotion ranking could be formed using an ordered list of emotions expressed by d, $(e_1, \ldots, e_m) \mid$ for $i, j \in (1, m)$, if $i < j$, then $d[e_i] > d[e_j]$, in which $d[e]$ is calculated using the lexicon as shown in Equation 15:

$$d[e] = \sum_{w \in d} \mathrm{Lex}(w, e) \times c(w, d). \quad (15)$$

Evaluation

We begin with the details of the benchmark datasets used in our evaluation, followed by results and discussion for perplexity analysis and lexicon quality assessment. We report the significance using a paired one-tailed t-test using 95 percent confidence.

Datasets

We conducted our comparative study of lexicons on four benchmark datasets. In our evaluation, we used 90 percent of the training data in each dataset for learning the lexicons and the remaining 10 percent as development data for parameter tuning (for example, MLE estimation of λ for the UMMs; we experimented with 11 values of λ (0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0) on each dataset for MLE). We used the test data in each dataset for perplexity analysis and lexicon quality assessment.

The news dataset (SemEval-2007; http://nlp.cs.swarthmore.edu/semeval/tasks/task14/summary.shtml) contains 1,250 emotional news headlines. We provided each headline with emotion ratings in the range [–100, 100] for Paul Ekman's basic emotions. We used this dataset for emotion ranking because it provides an ordered list of emotions on each news item.

The Twitter dataset is a collection of 2.6 million emotional tweets crawled from the Twitter search API using tweet identification numbers (http://knoesis.wright.edu/students/wenbo/download/dataset/twitter_emotion_SocialCom_Wang.tgz). We used the training dataset for learning DSELs in our comparative study, and we deployed the learned lexicons in the emotion-ranking task on a tweet event dataset.

The blog dataset (http://saimacs.github.io) contains 5,500 blog sentences annotated with Ekman's basic emotions. The dataset also includes words that reflect the sentence's emotion. Thus, we used this dataset to evaluate the quality of lexicons in predicting word-level emotions. We performed fivefold cross validation for our experiments, as opposed to tenfold, because the dataset was small.

The Emotion Event dataset contained 200 tweets describing emotional events (http://ahclab.naist.jp/resource/eped/data.zip). Each event is annotated with a ranked list of emotions by two annotators with agreement (kappa of 0.68). We used this dataset to test the quality of the lexicons on the emotion-ranking task. Because this dataset is very small, lexicons learned on the Twitter data were used here, because both datasets are crawled from Twitter.

Table 2. Word-emotion classification results for blogs.

Method	Average overall F-score (%)
Baseline GPELs	
WordNet-Affect (WNA)	29.96
NRC Emotion Lexicon (NRC)	39.05
EmoSenticNet (ESN)	28.30
Baseline DSELs	
Pointwise Mutual Information (PMI)	42.12
Word-Emotion Dictionary (WED)	24.51
Supervised latent Dirichlet allocation (sLDA)	38.72
Proposed DSEL	
UMM	**52.84**

Table 3. Document-emotion ranking results for news.

Method	Mean average precision (%)	Mean reciprocal rank (%)
Baseline DSELs		
PMI	64.66	30.53
WED	78.10	53.08
sLDA	67.44	35.42
Proposed DSEL		
UMM	**80.33**	**56.05**

Baselines and Metrics

Our comparative study includes baseline GPELs, such as WordNet-Affect (WNA); the NRC Emotion Lexicon and EmoSenticNet (ESN); baseline DSELs generated using Pointwise Mutual Information (PMI),[3] the Word-Emotion Dictionary (WED),[4] and sLDA[7]; and our proposed DSEL.

We assessed the DSELs' performance on both of the evaluation tasks, but GPELs can only be used for comparison in the word-emotion classification task, because they do not offer word-emotion quantifications needed for emotion ranking. In the word-emotion classification task, performance is reported using the standard metric F-score. For document-emotion ranking, we use mean reciprocal rank to measure the lexicon quality in predicting the dominant emotion present in the document, whereas we use the mean average precision to measure a lexicon's ability to order the multiple emotions connoted by a document.

Perplexity Analysis

Perplexity is a measure of how well an emotion language model θ_{e_k}, learned using the training data $D_{e_k}^{train}$, predicts the test (unseen) data $D_{e_k}^{test}$. Equation 16 shows how we calculate perplexity:

$$Perp\left(D_{e_k}^{test}\right) = 2^{-\frac{\sum_{i=1}^{\left|D_{e_k}^{test}\right|}\sum_{j=1}^{|d_i|}\log P\left(d_{ij}\middle|\theta_{e_k}\right)}{V_{e_k}}}, \quad (16)$$

in which V_{e_k} is the total number of words in the test data $D_{e_k}^{test}$. Therefore, the smaller the perplexity score, the better the language model is at predicting unseen data. Perplexity analysis is applied to UMM language models by considering values from the final EM iteration. Figure 2 shows the results for perplexity analysis on blogs, news, and tweets. UMM emotion topics had significantly lower perplexity than those of sLDA on all the three datasets, suggesting that the UMM is more effective than sLDA in capturing the documents' emotional characteristics.

Word-Emotion Classification Results

Table 2 shows word-classification results on blog data. The results are the average overall F-scores obtained over five folds. The proposed UMM lexicon performed significantly better than the GPELs (WNA, NRC, and ESN) and the baseline DSELs (PMI, sLDA, and WED). This evaluation clearly suggests that GPELs are generally inadequate for emotion detection because of their poor coverage of domain vocabulary. The assumption of DSELs such as WED and sLDA—that is, that documents

Table 4. Document-emotion ranking results for events.

Method	Mean average precision (%)	Mean reciprocal rank (%)
Baseline DSELs		
PMI	57.96	52.56
WED	50.07	46.30
sLDA	56.75	48.73
Proposed DSEL		
UMM	**61.7**	**57.27**

exhibit multiple emotions—proved to be less effective for predicting the emotion of a word in a context. PMI performed the best among the baselines by far; however, the proposed UMM's ability to penalize emotionally neutral words resulted in the best performance in predicting emotions at word level.

Document-Emotion Ranking Results

Tables 3 and 4 show the document-emotion ranking results for DSELs on news headlines and events captured by tweets, respectively. A comparison of the results for sLDA and WED lexicons on both the corpora suggest that they are more effective when the training documents exhibit multiple emotion characteristics, as in SemEval-2007. On the other hand, PMI gives better performance when documents exhibit single emotion characteristics, such as in tweets. However, the UMM's ability to quantify the emotionality and neutrality of words resulted in effective discrimination and ordering of document-level emotion associations across both the corpora.

In this article, we comparatively evaluated both GPELs and DSELs for emotion detection from text. Results from a comprehensive study of existing and proposed lexicons on emotion-detection tasks on benchmark datasets confirm that DSELs have significant performance gains over GPELs. Closer examination of DSEL results shows that the proposed lexicon outperformed those generated by state-of-the-art techniques such as PMI and sLDA in all emotion-detection tasks. A deeper empirical analysis suggests that the proposed method generates emotion language models (topics) that have significantly lower perplexity compared to those from sLDA. In the future, we plan to extend the proposed lexicon generation method to learn multiword-emotion lexicons (that is, bigram and trigram), following the recent trend in multiword sentiment and emotion detection.[8] We also plan to use the proposed DSEL's knowledge in conjunction with knowledge bases such as SenticNet and EmoSenticNet to extract effective features to represent documents for emotion classification. ■

References

1. A. Esuli, S. Baccianella, and F. Sebastiani, "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining," *Proc. Int'l Conf. Language Resources and Evaluation*, 2010, pp. 18–38.
2. C. Strapparava and A. Valitutti, "WordNet-Affect: An Affective Extension of Wordnet," *Proc. Int'l Conf. Language Resources and Evaluation*, 2004, pp. 1083–1086.
3. S.M. Mohammad, "#emotional Tweets," *Proc. 1st Jt. Conf. Lexical and Computational Semantics*, 2012, pp. 246–255.
4. Y. Rao et al., "Building Emotional Dictionary for Sentiment Analysis of Online News," *World Wide Web*, vol. 17, no. 4, 2014, pp. 723–742.
5. C. Yang, K.H.Y. Lin, and H.H. Chen, "Emotion Classification Using Web Blog Corpora," *Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence*, 2007, pp. 275–278.
6. M. Yang et al., "A Topic Model for Building Fine-Grained Domain-Specific Emotion Lexicon," *Proc. 52nd Ann. Conf. Assoc. Computational Linguistics*, 2014, pp. 421–426.
7. D.M. Blei and J.D. McAuliffe, "Supervised Topic Models," *Advances in Neural Information Processing Systems*, 2007, pp. 121–128.
8. E. Cambria, "Affective Computing and Sentiment Analysis," *IEEE Intelligent Systems*, vol. 31, no. 2, 2016, pp. 102–107.

Anil Bandhakavi is a PhD candidate in the Smart Information Systems research group at Robert Gordon University. Contact him at a.s.bandhakavi@rgu.ac.uk.

Nirmalie Wiratunga is a professor in the Smart Information Systems research group at Robert Gordon University. Contact her at n.wiratunga@rgu.ac.uk.

Stewart Massie is a research fellow in the Smart Information Systems research group at Robert Gordon University. Contact him at s.massie@rgu.ac.uk.

Deepak Padmanabhan is a lecturer in the School of Electronics, Electrical Engineering and Computer Science at Queen's University, Belfast. Contact him at deepaksp@acm.org.