Multi-Dimensional Sentiment Analysis with Learned Representations

Andrew L. Maas, Andrew Y. Ng, and **Christopher Potts**
Stanford University
Stanford, CA 94305
[amaas, ang, cgpotts]@stanford.edu

Abstract

Treating sentiment analysis as a classification problem has proven extremely useful, but it misses the blended, continuous nature of sentiment expression in natural language. Using data from the Experience Project, we study texts as distributions over sentiment categories. Analysis of the document collection shows the texts contain blended sentiment information substantially different from a categorization view of sentiment. We introduce a statistical vector-space model that learns from distributions over emotive categories, in addition to capturing basic semantic information in an unsupervised fashion. Our model outperforms several baselines in predicting sentiment distributions given only the text of a document.

1 Introduction

Computational sentiment analysis is often reduced to a classification task: each text is presumed to have a unique label summarizing its overall sentiment, and the goal is to build models that accurately predict those labels (Turney, 2002; Pang et al., 2002; Pang and Lee, 2008). The most widely-used labels are 'positive' and 'negative', with a third 'neutral' category also commonly included (Cabral and Hortaçsu, 2006). Sometimes this basic approach is enriched to a ranked or partially-ranked set of categories — for example, star ratings of the sort that are extremely common on the Web (Pang and Lee, 2005; Goldberg and Zhu, 2006; Snyder and Barzilay, 2007). And there is a large body of work employing other categories: not only binary distinctions like subjective vs. objective (Bruce and Wiebe, 1999; Wiebe et al., 1999; Hatzivassiloglou and Wiebe, 2000; Riloff and Wiebe, 2003; Riloff et al., 2005; Pang and Lee, 2004) and pro vs. con (Thomas et al., 2006), but also rich multidimensional category sets modeled on those of cognitive psychology (Liu et al., 2003; Alm et al., 2005; Wiebe et al., 2005; Neviarouskaya et al., 2010).

While treating sentiment as a classification problem is extremely useful for a wide range of tasks, it is just an approximation of the sentiment information that can be conveyed linguistically. The central assumption of the classification approach is that each text is uniquely labeled by one of the categories. However, human reactions are often nuanced, blended, and continuous (Russell, 1980; Ekman, 1992; Wilson et al., 2006). Consider, for example, this short 'confession' text from the website ExperienceProject.com:

> I have a crush on my boss! *blush* eeek *back to work*

At the Experience Project, users can react to texts by clicking buttons summarizing a range of emotions: 'sorry, hugs', 'that rocks', 'tee-hee', 'I understand', and 'wow, just wow'. At the time of this writing, the above confession had received the following distribution of reactions: 'that rocks': 1, 'tee-hee': 1, 'I understand': 10, and 'wow, just wow': 0. This corresponds well to the mix of human responses we might expect this text to elicit: it describes a socially awkward and complex situation, which provokes sympathetic reactions, but the text is light-

hearted in tone and thus likely to elicit less weighty rections as well. The comments on the confession reflect the summary offered by the reaction distribution: some users tease ("Ooooooooo.... i'm tellin!!! lol") and others offer encouragement ("you go and get that man...").

In this paper, we develop an approach that allows us to embrace the blended, continuous nature of human sentiment judgments. Our primary data are about 37,000 confessions from the Experience Project with associated reaction distributions. We focus on predicting those reaction distributions given only the confession text. This problem is substantially more challenging than simple classification, but we show that it is tractable and that it presents a worthwhile set of new questions for research in linguistics, natural language processing, and machine learning.

At the heart of our approach is a model that learns vector representations of words. The model has both supervised and unsupervised components. The unsupervised component captures basic semantic information distributionally. However, this document-level distributional information misses important sentiment content. We thus rely on our labeled data to imbue the word vectors with rich emotive information.

Visualization of our model's learned word representations shows multiple levels of word similarity (supplementary diagram A). At the macroscopic level, words are grouped into large clusters based on the reaction distributions they are likely to elicit, relfecting their sentiment connotations. Within these macroscopic clusters, words with highly related descriptive semantic content form sub-structures.

We evaluate our model based upon how well it predicts the reaction distributions of stories, but we also report categorization accuracy as a point of reference. To assess the impact of learning representations specifically for sentiment, we compare our model with several alternative techniques and find it performs significantly better in experiments on the Experience Project data.

2 Data

As noted above, our data come from the website ExperienceProject.com (EP). The site allows users to upload a variety of different kinds of texts, to comment on others' texts, and to contribute to annotating the texts with information about their reactions. We focus on the 'confessions', which are typically short, informal texts relating personal stories, attitudes, and emotions. Here are two typical confessions with their associated reactions:

> I really hate being shy ... I just want to be able to talk to someone about anything and everything and be myself... That's all I've ever wanted. [*understand*: 10; *hugs*: 1; *just wow*: 0; *rock*: 1; *teehee*: 2]

> subconsciously, I constantly narrate my own life in my head. in third person. in a british accent. Insane? Probably [*understand*: 0; *hugs*: 0; *just wow*: 1; *rock*: 7; *teehee*: 8]

Category	Clicks
'sorry, hugs'	22,236 (19%)
'you rock'	25,416 (22%)
'teehee'	16,052 (14%)
'I understand'	42,352 (37%)
'wow, just wow'	9,745 (8%)

Table 1: Overall distribution of reactions.

Our data consist of 37,146 texts (3,564,039 words; median text length of 56 words). Table 1 provides some basic information about the overall distribution of reactions. They are highly skewed towards the category 'I understand'; the stories are confessional, so it is natural for readers to be sympathetic in response. The 'wow, just wow' category is correspondingly little used, in virtue of the fact that it is largely for negative exclamation (its associated emoticon has its mouth and eyes wide open). Such reactions are reserved largely for extremely transgressive or shocking information.

We have restricted attention to the texts with at least one reaction. Table 2 summarizes the amount of reaction data present in this document collection, by measuring cut-offs at various salient points. When analyzing the reaction data, we normalize the counts such that the distribution over reactions sums to 1. This allows us to treat the reaction data as a

Reactions	Texts
$\geqslant 1$	37,146
$\geqslant 2$	24,179
$\geqslant 3$	15,813
$\geqslant 4$	10,537
$\geqslant 5$	7,073

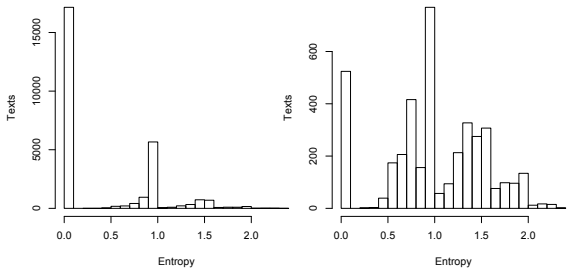
Table 2: Reaction counts.

probability distribution, ignoring differences in the raw number of counts stories receive.

There are many intuitive correlations between the authors' word choices and readers' reaction responses (Potts, 2010). Figure 1 illustrates this effect with words that show strong affinities to particular reaction types. Each panel depicts the distribution of the word across the rating categories. These were derived by first estimating $P(w|c)$, the probability of word w given class c, and then obtaining $P(c|w)$ by an application of Bayes rule under the assumption of a uniform prior over the classes. (Without this uniformity assumption, almost all words appear to associate with the 'understand' category, which is about four times bigger than the others; see table 1.) The gray horizontal line is at 0.20, the expected probability if there is no association between the word's usage and the reaction categories.

The first panel in figure 1 depicts *awesome*. As one might expect, this correlates most strongly with the 'rocks' and 'teehee' categories; stories in which one uses this word are likely to be perceived as positive and light-hearted (especially as compared to the usual EP fare). Conversely, *terrible*, in the second plot, correlates with 'hugs' and 'understand'; when an author describes something as terrible, readers react with sympathy and solidarity. The final panel depicts *cocaine*, one of a handful of words in the corpus that generate predominantly 'wow, just wow' reactions. We hope these examples help convey the nature of the reaction categories and also suggest that it is promising to try to use these data to learn sentiment-rich word vectors.

Finally, we address the question of how much the distributions matter as compared with a categorical view of sentiment. If the majority of the texts in the data received categorical or near-categorical responses, we might conclude that classification is an appropriate modeling choice. Conversely, if the texts tend to receive mixed reactions, then we are justified in adopting our more complex approach. Figure 2 assesses this using the entropy of the reaction distributions. Where the entropy is zero, just one category was chosen. Where the entropy is around two, the reactions were evenly distributed across the categories. As is evident from this plot, the overall picture is far from categorical; about one-third of the texts have a non-negligible amount of variation in their distributions. What's more, this picture is somewhat misleading. As table 2 shows, the majority of our texts have just one reaction. If we restrict attention to the 7,073 texts with at least five reactions, then the entropy values are more evenly distributed, with an entropy of zero far less dominant, as in figure 2(b). Thus, these texts manifest the blended, continuous nature of sentiment that we wish to model.



(a) The full corpus. (b) Texts with $\geqslant 5$ reactions.

Figure 2: The entropy of the reaction distributions.

3 Model

We introduce a model that captures semantic associations among words as well as the blended distributional sentiment information conveyed by words. We assume each word is represented by a real-valued vector and use a probabilistic model to learn words' vector representations from data. The learning procedure uses the unsupervised information of document-level word co-occurrences as well as the reaction distributions present in EP data.

Our work fits into the broad class of vector space models (VSMs), recently reviewed by Turney and Pantel (2010). VSMs capture word relationships by encoding words as points in a high-dimensional space. The models are both flexible and powerful;

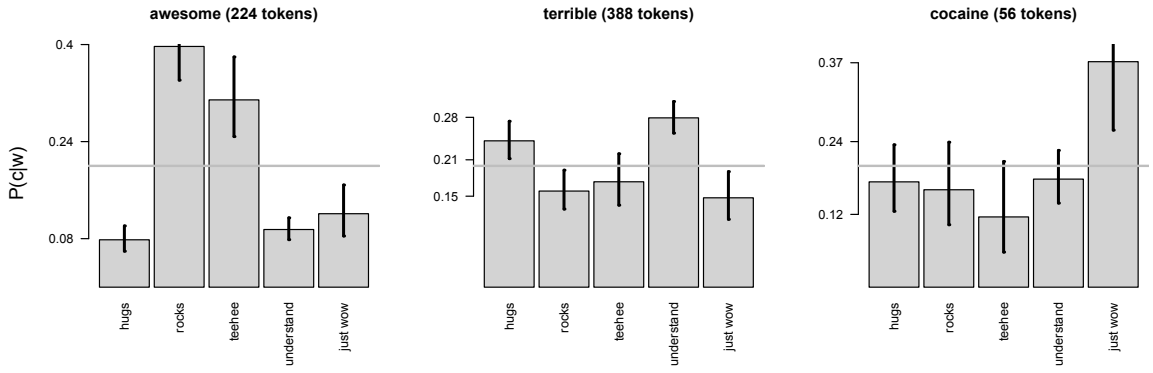


Figure 1: Word–category associations in the EP data.

depending on the application, the vector space can encode syntactic information, as is useful for named entity recognition systems (Turian et al., 2010), or semantic information, as is useful for information retrieval or document classification (Manning et al., 2008). Most VSMs apply some sort of matrix factorization technique to a term-context co-occurrence matrix. However, the success of matrix factorization techniques for word vectors often depends heavily on the choices one makes for weighting the entries (for example, with inverse document frequency of words). Thus, the process of building a VSM requires many design choices, often with only past empirical results as guidance. This challenge is multiplied when building representations for sentiment because we want word vectors to capture both descriptive and emotive meanings. The recently introduced delta inverse document frequency weighting technique has had some success in binary sentiment categorization (Martineau and Finin, 2009), but it does not naturally handle multi-dimensional notions of sentiment.

Our recent work seeks to address these design issues. In Anonymous (2011), we introduce a probabilistic model for learning semantically-sensitive word vectors. In the present paper, we build off of this probabilistic model of documents, because it helps avoid the large design space present in matrix factorization-based VSMs, but we extend its sentiment component considerably. Whereas we previously learned only from unique labels, we are now able to capture the multi-dimensional, non-categorical notion of sentiment that is expressed in the EP data. In the following sections, we introduce the semantic and sentiment components of the model separately, and then describe the procedure for learning the model's parameters from data.

3.1 Semantic Component

We approximately capture word semantics from a collection of documents by analyzing document-level word co-occurrences. This semantic component uses a probabilistic model of a document as introduced in our previous work (Anonymous, 2011). The model uses a continuous mixture distribution over words indexed by a multi-dimensional random variable θ. Informally, we can think of each dimension of a word vector as a topic in the sense of topic modeling. The document coefficient vector θ thus encodes the strength of each topic for the document. A word's probability in the document then corresponds to how strongly the word's topic strengths match those defined by θ.

We assign a probability to a document d using a joint distribution over the document and θ. The model assumes each word $w_i \in d$ is conditionally independent of the other words given θ, a bag of words assumption often used when learning from document-level co-occurrences. The probability of a document is thus,

$$p(d) = \int p(d, \theta) d\theta = \int p(\theta) \prod_{i=1}^{N} p(w_i|\theta) d\theta, \quad (1)$$

where N is the number of words in d and w_i is the

i^{th} word in d.

The conditional distribution $p(w_i|\theta)$ is defined using a softmax distribution,

$$p(w|\theta; R, b) = \frac{\exp(\theta^T \phi_w + b_w)}{\sum_{w' \in V} \exp(\theta^T \phi_{w'} + b_{w'})}. \quad (2)$$

The parameters of the model are the word representation matrix $R \in \mathbb{R}^{(\beta \text{ x } |V|)}$ where each word w (represented as a one-on vector) in the vocabulary V has a β-dimensional vector representation $\phi_w = Rw$ corresponding to that word's column in R. The random variable θ is also a β-dimensional vector, which weights each of the β dimensions of words' representation vectors. A scalar bias b_w for each word captures differences in overall word frequencies. The probability of a word given the document parameter θ corresponds to how strongly that word's vector representation ϕ_w matches the scaling direction of θ.

Equation 1 resembles the probabilistic model of latent Dirichlet allocation (LDA) (Blei et al., 2003), which models documents as mixtures of latent topics. However, our model does not attempt to model individual topics, but instead directly models word probabilities conditioned on the topic mixture variable θ. Our previous work compares the word vectors learned with our semantic component to an approach which uses LDA topic associations as word vectors. We found the word vectors learned with our model to be superior in tasks of document and sentence-level sentiment classification.

Maximum likelihood learning in this model assumes documents d_k in a collection D are i.i.d. samples. The learning problem of finding parameters to maximize the probability of observed documents becomes,

$$\max_{R,b} p(D; R, b) = \prod_{d_k \in D} \int p(\theta) \prod_{i=1}^{N_k} p(w_i|\theta; R, b) d\theta. \quad (3)$$

Using *maximum a posteriori* (MAP) estimates for θ, we approximate this learning problem as,

$$\max_{R,b} \prod_{d_k \in D} p(\hat{\theta}_k) \prod_{i=1}^{N_k} p(w_i|\hat{\theta}_k; R, b), \quad (4)$$

where $\hat{\theta}_k$ denotes the MAP estimate of θ for d_k. Our previous work used a Gaussian prior for θ. In our present experiments we explore both Gaussian and Laplacian priors. The Laplacian prior is intuitively appealing because it encourages sparsity, where certain entries of θ are exactly zero as opposed to small non-zero values as is the case when using Gaussian priors. These exactly zero values correspond to topic dimensions which are not at all present in the semantic representation of a word.

3.2 Sentiment Component

We now introduce the second component of our model, which aims to capture the multi-dimensional sentiment information expressed by words. Unlike topical information, sentiment is not easy to learn by analyzing document-level word co-occurrences alone. For this reason, we use the reaction distributions of documents to capture how words in the document express multi-dimensional sentiment information. Our previous work demonstrated the value of learning sentiment-sensitive word representations for the simplistic binary categorization notion of sentiment. We now introduce a method to learn word vectors sensitive to a continuous multi-dimensional notion of sentiment.

Our model dictates that a word vector ϕ should predict the reaction distribution of documents in which that word occurs using an appropriate predictor function. Because the reaction distributions are categorical probability distributions, we use a softmax model,

$$\hat{s}_k = \frac{\exp(\psi_k^T \phi + c_k)}{\sum_{k'} \exp(\psi_{k'}^T \phi + c_{k'})} \quad (5)$$

The value $\hat{s}_k$ is the probability predicted for the k^{th} sentiment dimension for a given word vector ϕ. The softmax weight vectors ψ_k serve to partition the vector space into K regions where each region corresponds to a particular sentiment dimension. The predicted reaction distribution for a word thus depends on where that word lies in the vector space relative to the regions defined by ψ.

For EP data, a document d is associated with its reaction distribution s, which is a five-dimensional categorical probability distribution ($K = 5$). The softmax parameters $\psi \in \mathbb{R}^{K \text{ x } \beta}$ and $c \in \mathbb{R}^K$ are

shared across all word vectors as to create a single set of emotive regions in the word vector space. The softmax predicts a reaction distribution for each word, and we learn the softmax parameters as well as the word vectors to match the observed reaction distributions.

The predicted and actual reaction distributions are categorical probability distributions, so we use the Kullback-Leibler (KL) divergence as a measure of how closely the predicted distribution matches the actual. Given the actual distribution s and a prediction for this distribution $\hat{s}$ the KL divergence is,

$$KL(\hat{s}||s) = \sum_{k=1}^{K} s_k \log \frac{s_k}{\hat{s}_k}. \quad (6)$$

Learning this component of the model amounts to finding word vectors as well as softmax parameters to minimize the KL divergence between reaction distributions of observed documents and the predicted reaction distributions of words occurring in the documents. We can formally express this as,

$$\min_{R,\psi,c} \sum_{d_k \in D} \sum_{i=1}^{N_k} KL(\hat{s}^{w_i}||s), \quad (7)$$

where $\hat{s}^{w_i}$ is the predicted reaction distribution for word w_i as computed by (5). To ensure identifiability of the softmax parameters ψ we constrain $\psi_K = 0$

3.3 Learning

We now describe the method to learn word vectors using both the semantic and sentiment components of the model. The learning procedure for the semantic component minimizes the negative log of the likelihood shown in equation (4). The sentiment component is then additively combined to form the full learning problem,

$$\min_{R,b,\psi,c} \lambda||R||_F^2 + \sum_{d_k \in D} \sum_{i=1}^{N_k} KL(\hat{s}^{w_i}||s) - \left(\sum_{k=1}^{|D|} \log p(\hat{\theta}_k) + \sum_{i=1}^{N_k} \log p(w_i|\hat{\theta}_k; R, b) \right). \quad (8)$$

We add to the objective Frobenious norm regularization on the word representation matrix R to prevent the word vector norms from growing too large. We minimize the objective function for several iteration using the L-BFGS quasi-Newton algorithm while leaving the MAP estimates $\hat{\theta}$ fixed. The MAP estimates are then updated while leaving the other parameters of the model fixed. This process continues until the objective function value converges.

Our work explores both a Gaussian and a Laplacian prior for θ. The log-Gaussian prior corresponds to a squared ℓ_2 (sum of squares, $\sum_i x_i^2$) penalty on θ whereas the Laplacian prior corresponds to an ℓ_1 (sum of absolute values,$\sum_i |x_i|$) penalty. Both priors have a single free parameter λ which is proportional to the variance of the prior distribution. This regularization parameter λ and the word vector dimensionality β are the only free hyper-parameters of the model. Because optimizing the non-differentiable ℓ_1 penalty is difficult with gradient-based techniques we approximate the ℓ_1 penalty with the function $\log \cosh(\theta)$.

4 Experiments

Our experiments focus on predicting the reaction distribution given the text of a document. We employ several baseline approaches to assess the relative performance of our model. As shown in figure 2, the reaction distributions of stories which received at least five reactions have higher entropy on average than the set which includes stories with only one reaction or more. The higher entropy reaction distributions are of greater interest because predicting such distributions is substantially more challenging than predicting a low entropy distribution, which is more like the categorization approach of previous work. We evaluate models on both the set of texts with at least one reaction, and the set of texts with five or more reactions.

After collecting the text and reaction distributions from the Web, we tokenized all documents with at least one reaction. Traditional stop word removal was not used because certain stop words (e.g. negations) are indicative of sentiment. To minimize the amount of text pre-processing, we did not apply stemming or spelling correction. Because certain non-word tokens (e.g. "!" and ":-)") are indicative of sentiment, we allow them in our vocabulary. After this tokenization, the dataset consists of 52,973 unique unigrams, many of which occur only once

because they are unique spellings of words (e.g. "hahhhaaa"). The collection of 37,146 documents is reduced to 37,130 when we discard documents with no tokens recognized by our tokenizer. Most stories fall around the median length of 56 words, however, a few are thousands of words long. We randomly partitioned the data into 30,000 training and 7,130 test documents. When we consider documents with at least five reactions, this becomes 5,764 training and 1,307 test documents.

4.1 Word Representation Learning

We induce word representations with our model using the learning procedure described in section 3.3. We construct word representations for only the 5,000 most frequent tokens in the training data. This speeds computation and avoids learning uninformative representations for rare words for which there is insufficient data to properly assess their semantic and sentiment associations. We use the 29,591 documents from our training set with length at least five when the vocabulary is restricted to the 5,000 most frequent tokens. The reaction distributions for documents are used when learning the sentiment component of the model. Our model could leverage additional unlabeled data from related websites to better capture the semantic associations among words. However, we restrict the model to learn from only the labeled training set in order to better compare it to baseline models for this task.

For both the Gaussian and Laplacian models, we evaluate 100-dimensional word vectors and set the regularization parameter $\lambda = 10^{-4}$. Our previous work and preliminary experiments with this dataset suggested the learned word vectors are relatively insensitive to changes in these parameters.

Supplementary diagram A shows a 2-D visualization of the learned word similarities for the 2,000 most frequent words in our vocabulary. The visualization was created using the t-SNE algorithm, with code provided by van der Maaten and Hinton (2008). Word vectors are cosine normalized before passing them to the t-SNE algorithm.

The visualization clearly shows words grouped locally by semantic associations — for example, "doctor" and "medication" are nearby. Additionally, there is some evidence that the macroscopic structure of the words correlates with how they influence reaction distributions. A cluster of words containing playful, upbeat tokens like ":-)" and "haha" are all likely to appear in stories which elicit the *rock* or *tee-hee* reactions. Far removed from such happy words are clusters of words indicative of melancholic subjects, marked by words like "cancer" and "suicide." We note that sad and troubling topics are highly prevalent in the data, and our visualization reflects this fact.

After learning the word representations, we represent documents using *average word vectors*. This approach uses the arithmetic average of the word vectors for all words which appear in the document. Because we learn word vectors for only the 5,000 most frequent words, a small fraction of the documents contain only words for which we do not have vector representations. These documents are represented as a vector of all zeros.

4.2 Alternative Methods

In addition to the vectors induced using our model, we evaluate the performance of several standard approaches to document categorization and information retrieval.

Unigram Bag of Words Representing a document as a vector of word counts performs surprisingly well in many classification tasks. In our preliminary experiments, we found that term presence performs better than term frequency on EP data, as noted in previous work on sentiment (Pang et al., 2002). We also note that delta inverse document frequency weighting, which has been shown to sometimes perform well in sentiment (Martineau and Finin, 2009), does not extend easily to multi-dimensional notions of sentiment. We thus use term presence vectors with no normalization and evaluate with the full vocabulary of the dataset and the 5,000 word vocabulary used in building word vectors.

Latent Semantic Analysis (LSA) We apply truncated singular value decomposition to a term-document count matrix to obtain word vectors from LSA (Deerwester et al., 1990). We first apply tf.idf weighting to the term-document matrix, but do not use cosine normalization. We use the same 5,000 word vocabulary as is used when constructing word vectors for our model.

Features	$\geqslant 5$ reactions KL	$\geqslant 5$ reactions Max Acc.	$\geqslant 1$ reaction KL	$\geqslant 1$ reaction Max Acc.
Uniform Reactions	0.861	20.2	1.275	20.4
Mean Training Reactions	0.763	43.0	1.133	46.7
Bag of Words (All unigrams)	0.637	56.0	1.000	53.4
Bag of Words (Top 5000 unigrams)	0.640	54.9	0.992	54.3
LSA	0.667	51.8	1.032	52.2
Our Method Laplacian Prior	0.621	55.7	0.991	54.7
Our Method Gaussian Prior	0.620	55.2	0.991	54.6

Table 3: Test set performance.

4.3 High Entropy Reaction Distributions

Our first experiment considers only the examples with at least five reaction clicks, because they best exhibit the blended distributional notion of sentiment of interest in this work. For all of the feature sets described (mean word vectors and bag of words), we train a softmax classifier on the training set. The softmax classifier is a predictor of the same form as is described in equation (5), but with a quadratic regularization penalty on the weights. The strength of the regularization penalty is set by cross-validation on the training set. The classifier is trained to minimize the KL divergence of predicted and actual distributions on the training set. We then evaluate the models by measuring average KL divergence on the test set.

We also report performance of models in terms of accuracy in predicting the maximum probability reaction for a document. In this setting, the model picks a single category corresponding to its most probable predicted reaction. A prediction is counted as correct if that category is the most probable in the true reaction distribution, or if it is tied with other categories for the role of most probable. None of the models were explicitly optimized to perform this task, but instead to predict the full distribution of reactions. However, it is helpful to compare this performance metric to KL divergence, as measuring performance in terms of accuracy is more familiar. Table 3 shows the results; recall that lower average KL divergence indicates better performance.

All bag of words and vector space models beat the simplistic baselines of predicting the average reaction distribution, or a uniform distribution. The improvements in both KL divergence and accuracy are substantial relative to these simplistic baselines, suggesting that it is indeed feasible to predict reaction distributions from text. Both variants of our model perform better than bag of words and LSA in KL divergence, but bag of words performs best using the accuracy as the metric. That the accuracy and KL metrics disagree on models' performance rankings suggests categorization accuracy is not a sufficient indicator of how well models capture a distributional notion of sentiment. Based on the poor performance of LSA-derived word vectors, we hypothesize that learning representations using sentiment distributions is critical when attempting to capture the blended sentiment information within documents.

Differences in KL divergence are somewhat difficult to interpret, so we use a matched t-test to evaluate their significance. The matched t-test between two models takes the KL divergence for each test example and evaluates the hypothesis that the KL divergence numbers come from the same distribution. KL divergences on the set of test examples are approximately gamma distributed with a valid range of $[0, \infty]$. We thus apply the matched t-test to the logarithm of the KL divergences, which have a Gaussian distribution as assumed by the t-test. We find that the difference in KL divergence between our models and the bag of words models are significant ($p < 0.001$). However, the Gaussian and Laplacian prior variants of our model do not differ significantly from each other. The prior over document coefficients perhaps has little effect relative to the other components of our model, causing both

model variants to perform comparably.

4.4 All Reaction Distributions

We repeated the experimental procedure using the full dataset which includes all documents with at least one reaction. As noted in figure 2, these reaction distributions have low average entropy because a large number of documents have only a few reactions. Distribution predictors for all models were trained and evaluated on this dataset; table 3 shows the results.

Again all models outperform the naive baselines of guessing the average training distribution or a uniform distribution. A third baseline (not show) which assigns 99% of its probability mass to the dominant *understand* category performs substantially worse than all results shown. Although the difference in KL divergence between our models and the bag of words baselines are numerically small, the improvement of our models is significant as measured by the matched t-test ($p < 0.001$). The significance of such small differences is due to the large testing set size. Again the Gaussian and Laplacian variants of our model do not differ significantly from each other in performance.

We see that all models have a higher average KL divergence on this task as compared to evaluation on the set of documents with at least five reactions. As shown in table 2, reaction distributions with zero entropy dominate this version of the dataset. We hypothesize that the higher average KL divergences and small numerical differences in KL divergence are largely due to all predictors struggling to fit these zero entropy distributions which were formed with only one reaction click.

5 Conclusion

Using the confessions at the EP, we showed that natural language texts often convey a wide range of sentiment information to varying degrees. While classification models can capture certain emotive dimensions, they miss this blended, continuous nature of sentiment expression. Building on the existing classifier model of Anonymous (2011), we developed a vector-space model that learns from distributions over emotive categories, in addition to capturing basic semantic information in an unsupervised fashion. The model is successful in absolute terms, suggesting that learning realistic sentiment distributions is tractable, and it also outperforms various baselines, including LSA. We believe the task of predicting sentiment distributions from text provides a rich challenge for the field of sentiment analysis, especially when compared to simpler classification tasks. Going forward, we plan to move beyond the lexical level to capture the ways in which sentiment is influenced by compositional semantic facts (e.g., interaction with negation and other non-veridical operators), which we expect to provide further insights into the complexities of sentiment expression.

Acknowledgments

This work is supported by the DARPA Deep Learning program under contract number FA8650-10-C-7020, an NSF Graduate Fellowship awarded to AM, ONR grant No. N00014-10-1-0109, and ARO grant No. W911NF-07-1-0216.

References

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: Machine learning for text-based emotion prediction. In *Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(4-5):993–1022, May.

Rebecca F. Bruce and Janyce M. Wiebe. 1999. Recognizing subjectivity: A case study in manual tagging. *Natural Language Engineering*, 5(2).

Luís Cabral and Ali Hortaçsu. 2006. The dynamics of seller reputation: Theory and evidence from eBay. Working paper, downloaded version revised in March.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, September.

Paul Ekman. 1992. An argument for basic emotions. *Cognition and Emotion,*, 6(3/4):169–200.

Andrew B. Goldberg and Jerry Zhu. 2006. Seeing stars when there aren't many stars: Graph-based semi-supervised leaarning for sentiment categorization. In *TextGraphs: HLT/NAACL Workshop on Graph-based Algorithms for Natural Language Processing*.

Vasileios Hatzivassiloglou and Janyce Wiebe. 2000. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the International Conference on Computational Linguistics (COLING)*.

Hugo Liu, Henry Lieberman, and Ted Selker. 2003. A model of textual affect sensing using real-world knowledge. In *Proceedings of Intelligent User Interfaces (IUI)*, pages 125–132.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition.

J. Martineau and T. Finin. 2009. Delta tfidf: An improved feature space for sentiment analysis. In *Proceedings of the third AAAI internatonal conference on weblogs and social media*.

Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka. 2010. Recognition of affect, judgment, and appreciation in text. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 806–814, Beijing, China, August. COLING 2010 Organizing Committee.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 271–278.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1):1–135.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86, Philadelphia, July. Association for Computational Linguistics.

Christopher Potts. 2010. On the negativity of negation. In David Lutz and Nan Li, editors, *Proceedings of Semantics and Linguistic Theory 20*. CLC Publications, Ithaca, NY.

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Ellen Riloff, Janyce Wiebe, and William Phillips. 2005. Exploiting subjectivity classification to improve information extraction. In *Proceedings of AAAI*, pages 1106–1111.

James A. Russell. 1980. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.

Benjamin Snyder and Regina Barzilay. 2007. Multiple aspect ranking using the Good Grief algorithm. In *Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference (HLT-NAACL)*, pages 300–307.

Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of EMNLP*, pages 327–335.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the ACL*.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188.

Peter Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 417–424.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, November.

Janyce M. Wiebe, Rebecca F. Bruce, and Thomas P. O'Hara. 1999. Development and use of a gold standard data set for subjectivity classifications. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 246–253.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation (formerly Computers and the Humanities)*, 39(2/3):164–210.

Theresa Wilson, Janyce Wiebe, and Rebecca Hwa. 2006. Just how mad are you? Finding strong and weak opinion clauses. *Computational Intelligence*, 2(22):73–99.