Personalized Emotion Classification with Latent Dirichlet Allocation

Kui Tang
Columbia University, New York, NY 10027, USA
{kt2384}@columbia.edu,

Abstract

The inability for machines to infer emotion handicaps user interaction in all applications and limits accuracy of applications that monitor human behavior, such as anti-fraud systems. Related work centers on one-dimensional representations of emotion as well as time-series analysis of aggregate sentiment of many people, both of which offer limited utility to a designer of emotionally-aware applications. In this project, we derive a personal, probabilistic emotional lexicon from Tumblr blogs. This lexicon can then be used to infer future emotional state. We take a semi-supervised approach with latent Dirichlet allocation (LDA) and a derivative, SubjLDA, initializing our model with labeled emotional words with a small non-domain-specific lexicon and allowing the model to classify the remaining words and documents into emotional categories.

1 Introduction

> It is obvious, that when we have the prospect of pain or pleasure from any object, we feel a consequent emotion of aversion or propensity, and are carryed to avoid or embrace what will give us this uneasines or satisfaction. . . . Here then reasoning takes place to discover this relation; and according as our reasoning varies, our actions receive a subsequent variation. But it is evident in this case that the impulse arises not from reason, but is only directed by it.
>
> — Hume [6]

As Humean are ultimately motivated by emotion, applications concerned with user behavior, from recommender engines to email clients to anti-fraud systems find the need for knowledge of a user's emotional state. Here, we confront a basic problem of inferring a user's emotional state from text.

Given a set of blog posts from one user, we wish to learn a personalized, probabilistic lexicon of emotion. Certain words such as 'love' and 'fear' clearly signify a particular emotion. However, manually curated lexica are sparse, and each person uses language in a slightly different way and carries unique connotations. A medical student write about her work experiences at the hospital, thereby associating the word 'hospital' with aspiration and career success. But most people associate hospitals with fear and uncertainty. Thus, in a similar situation as speech recognition, applications that attempt to infer the user's emotional state should improve their performance when trained user-specific data.

In this project, we start with a small, hand-labeled lexicon of emotional words as a baseline. We compare this baseline against two topic models adapted for emotional classification and show significant improvements against baseline.

1.1 Related Work

The explosion of social media—particularly (micro)blogs, product reviews, and recommendations—drives both demand and opportunity for mining sentiments from text. Early work by Turney in 2002 [13] uses an unsupervised method of computing *pointwise mutual information* between words in product reviews with the seed words "excellent" and "poor". Indeed, most work, including the model on which this project is based [7], models emotion as a one-dimensional continuum between positive and negative.

We thus move to multidimensional emotional analysis. Rather than start with merely two mood states, multidimensional analyses begin with lists of seed words for each emotional dimension of interest. The earliest standard lexicon is the the art is the *Profile of Mood states* [8], a psychological survey that includes 65 items labeled across six distinct states. Bollen et al. extend this list to a lexicon of 964 terms by adding co-related words appearing in Google's n-gram data and use this data to regress DJIA closing prices against a time series of aggregate inferred Twitter sentiment [5]. They find more significant Granger-causality p-values in a lag of 2–6 days when incorporating a multidimensional lexicon compared to significance only in one-day lags on a unidimensional lexicon. The WordNet-Affect project begins with a seed list of manually curated terms, and then iteratively finds synsets with WordNet to expand the list [12]. In this project we use WordNet-Affect data as our priors.

Texts do not generally come with explicit emotional tags, so supervised techniques have limited scope. One notable exception is Livejournal, which allows users to specify one of 142 distinct moods when composing a blog post. Mishne et. al [10] use these labels to build a linear regression model, predicting mood *levels*—the sum of mood tags at a given hour—from total post intensity, hour of the day, and frequencies of individual terms. Sood and Vasserman use k-means to compactify the set of 142 mood descriptors into just 3 (happy, sad, angry) and apply Naive Bayes with the new labels.

2 Models

2.1 Latent Dirichlet Allocation

Latent Dirichlet allocation [4] models each document d (here, a blog post) as a multinomial distribution θ_d over a sparse subset of latent topics Z. Each index i in a document is drawn from this multinomial topic distribution, giving each word a topic label z_i^d. Conditioned on this topic label, a word w_i^d is drawn from a multinomial distribution $\beta_{z_i^d}$. Here, we fix the number of topics to 7 to include the six emotions in our prior lexicon plus a neutral emotion. We implement the standard collapsed Gibbs sampler.

2.2 SubjLDA

SubjLDA is a model by Lin [7] which adds a sentence-level subjectivity latent variable. Each sentence is either objective or subjective. Conditioned on subjectivity, a sentiment label is drawn for each word in that sentence. Since a single document can shift emotions, the sentence-level hierarchy allows this model to better capture local sentiment. The generative procedure for SubjLDA follows:

1. For each sentiment label l:
   (a) Draw $\phi_l \sim \text{Dir}(\lambda_l \cdot \beta_l)$, a multinomial distribution over words for the sentiment label l.
2. For each document d:
   (a) Draw $\pi_d \sim \text{Dir}(\gamma)$, a multinomial distribution over subjectivity labels for each sentence in document d.
   (b) For each sentence j:
      i. Draw a subjectivity label $s_{d,j} \sim \text{Multinomial}(\pi_d)$
      ii. Draw a $\theta_{d,m} \sim \text{Dir}(\alpha_{\mathbf{s_{d,m}}})$, a distribution of sentiments for sentence j of document d.
      iii. For each $N_{d,m}$ word position in sentence m of document d

A. Draw a sentiment $l_{d,m,t} \sim \text{Mult}(\theta_{s_{d,m}})$
B. Draw a word $w_{d,m,t} \sim \text{Mult}(\phi_{l_{d,m,t}})$

Each α encodes an asymmetric Dirichlet prior on the distributions of sentiments given a subjectivity. In our inference, we initialize our labels with a training corpus, and use a maximum-likelihood estimate for α, computed using a Newton-Raphson iteration from the initializing sentiment and subjectivity labels [9]. Each β is scaled to the empirical frequency of training words for each sentiment. The prior γ controls the mixing coefficients of subjectivity labels (subjective or objective). Thus, our priors capture the asymmetry of sentiment: that objective and subjective sentences are not equally prevalent, that most words are neutral, and that words may not be distributed uniformly across emotions. Further details on these priors below.

Let $\mathbf{w}$ denote a bag-of-words vector of words, $\mathbf{s}$ denote a vector of subjectivity labels, and $\mathbf{l}$ denote a vector of sentiment labels. Then the joint distribution is

$$\begin{aligned} P(\mathbf{w}, \mathbf{s}, \mathbf{l}|\alpha, \beta, \gamma) &= P(\mathbf{w}|\mathbf{l}, \beta)P(\mathbf{l}|\mathbf{s}, \alpha)P(\mathbf{s}|\gamma) \\ &= \int P(\mathbf{w}|\mathbf{l}, \beta)P(\mathbf{l}|\beta)d\Phi \int P(\mathbf{l}|\mathbf{\Theta})P(\mathbf{\Theta}|\mathbf{s}, \alpha)d\Theta \int P(\mathbf{s}|\mathbf{\Pi})P(\mathbf{\Pi}|\gamma)d\mathbf{\Pi} \end{aligned} \tag{1}$$

We can analytically integrate out the parameters to get conditional distributions for each sentiment and subjectivity label and thus build a Gibbs sampler. For details, see [7].

2.3 Priors

We follow Lin's approach to implement semi-supervised learning, initializing the latent variables and model priors using existing corpora. We use the words under the EMOTION label in WordNet-Affect 1.0 [12], which provides 606 words labeled with one of "anger", "disgust", "fear", "joy", "sadness", "surprise". We also used the neutral words from SentiWordNet [2], the same source as Lin.

For each word in the training set, if we found that word in either prior lexicon, we set the appropriate sentiment label. If a sentence contained any non-neutral words, we labeled that sentence subjective. Otherwise, we labeled the sentence objective.

If a word was not matched by emotion or neutrality (rare; less than 10%) of the time, we randomly assigned it an emotion.

The majority of words were neutral. Of 286,997 words in the poetry blog, 240,087, or 83.6%, were discovered to be neutral.

3 Data

Data were scraped from Tumblr in April 2012. The API was not usable because it only allowed access to users who have authenticated the API key for this application. However, as this model works on only a single user's blog, applications could work over the API.

Not all Tumblr sites shared the same HTML layout, but upon inspection of the source codes, a small number of XPath queries were able to extract all textual content. Parsing and text extraction was performed with lxml.

For these results, we analyzed three blogs, consisting mainly of personal writings in politics, prose, and poetry. All posts were by the same user, a critical feature of our model. summarizes our data.

Because many Tumblr posts are short or contain only images or links, they are not suitable for lexical analysis. Thus, we dropped all posts with fewer than 20 words. Table 1 gives a sense for the scope of our data, after cleaning. We randomly chose 20% of posts to hold as a test set to evaluate our model's performance.

URL	Genre	Posts	Words	Vocabulary
thenatureofdaylight.tumblr.com	poetry	1,325	88,664	8,973
noconsensusonaname.tumblr.com	prose	288	26,090	5,958
pantslessprogressive.com	politics	4,736	360,886	28,592

Table 1: Sizes of our data sets, after stemming and removing stopwords and short posts.

3.1 Feature Extraction

Both LDA and SubjLDA represent the document as a bag of words. In addition, SubjLDA requires identifying sentences in documents, as it posits that sentences are associated with distinct distributions of sentiment.

To parse the data into sentences and words, we used the standard Punkt sentence tokenizer in NLTK [3]. To reduce data sparsity, we ran the Porter stemmer and removed stemmed stopwords.

4 Results

A simple single-core Python implementation of Gibbs sampling for SubjLDA processed on average 2,715.8 words per second, which comes to 26.0 seconds per iteration for the poetry blog of 1325 posts. For each of the three blogs, we ran 400 Gibbs sampling iterations of both LDA and SubjLDA. The largest dataset, pantslessprogressive.com, over 24 hours to reach 400 iterations. No effort was made to diagnose convergence, though there should have been.

4.1 Highly Probable Words

Table 2 lists the most frequently occurring words for each emotion. The words appearing in WordNet-Affect are bolded. The prevalence of non-bold words indicate that the models learned from the data.

In particular, the politics blog, pantslessprogressive.com, is over four times larger than the other dataset, and consequently contains fewer bolded words, showing on this particular corpus, the learned emotional words were more significant than those from the prior lexicon.

Quite a few words have their labels flipped. This occurs when a blogger uses a word in contexts different from what the lexicon creators anticipated. For instance, in both models, "expect" shows up in the SADNESS column for the blog nonconsensusonaname. Indeed, themes on this blog are quite grim, suggesting that most expectations are negative. The WordNet-Affect creators categorized "expect" as JOY. Clearly not everybody is optimistic as they are, exemplifying the need for personalized sentiment analysis.

Furthermore, the model successfully chose sparse topic-word and document-topic representations. Figure 1 shows the word mixing proportions for each sentiment for the blog noconsensusonaname.tumblr.com.

4.2 Evaluation

For each model, we report the perplexity of the test set [4]. Intuitively, if probability model assigns high probabilities to newly observed events, then the exponent term is large, so the perplexity is low. The model is less "surprised" by the new information—in fact, it is more likely to predict this new information. Since perplexity is evaluated on a test set, lower perplexity implies better generalization.

$$perp(D_{test}) = \exp\left(-\frac{\sum_{d=1}^{M}\log p(\mathbf{w}_d)}{\sum_{d=1}^{N} N_d}\right) \quad (2)$$

where $\mathbf{w}_d$ denotes the sequence of words for document d.

Our baseline is the labels from WordNet-Affect and SentiWordNet alone. We must formulate this baseline as a generative model in order to compute perplexity. To do so, we allow the baseline a trivial training step to label each sentiment word in training corpus as it is able to, and then for each

SubjLDA on thenatureofdaylight.tumblr.com

NEUTRAL	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE
life	re	redrum	n't	**love**	n't	ask
one	**get**	sourc	**like**	**like**	know	**friend**
time	**look**	day	ca	**give**	m	anonym
thing	fuck	jonathan	want	would	**like**	**get**
want	make	grow	**live**	everyth	**feel**	know
never	**good**	**enjoy**	question	alway	even	**like**
think	say	definit	**away**	**look**	go	best
everi	n't	fight	oh	**live**	**sad**	re
feel	guy	bridg	**dark**	make	miss	n't
peopl	**hate**	perhap	promis	person	**pleas**	girl

LDA on thenatureofdaylight.tumblr.com

NEUTRAL	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE
n't	**see**	redrum	**away**	**love**	fog	wind
like	fuck	**love**	**heart**	life	fuck	cock
want	**love**	**away**	**air**	**give**	**along**	rose
know	**hate**	region	**shadow**	**heart**	bus	stand
one	amp	link	**like**	**see**	**tear**	templ
m	reggi	everyon	**see**	**live**	hors	**heart**
time	tito	**scene**	color	**tear**	**away**	begin
feel	line	book	sourc	**away**	lost	**air**
thing	**red**	point	paint	**scream**	**like**	dead
would	lyric	**short**	band	**inspir**	chandeli	**get**

SubjLDA on noconsensusonaname.tumblr.com

NEUTRAL	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE
one	cours	**like**	someth	**like**	would	n't
know	**short**	would	tri	**look**	want	**get**
could	believ	**turn**	**away**	n't	everyth	re
go	began	never	**emot**	fuck	**dark**	anyth
make	lie	mouth	import	**live**	**expect**	**good**
back	cool	d	mayb	**move**	**blue**	yet
even	count	shoulder	sentenc	**walk**	notic	liam
peopl	alec	**good**	music	**friend**	**bad**	understand
hand	**class**	fall	stand	peopl	mean	ca
time	luck	none	human	could	mark	conrad

LDA on noconsensusonaname.tumblr.com

NEUTRAL	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE
n't	build	milo	**get**	**look**	**blue**	**get**
one	two	vin	**good**	**move**	**dark**	**good**
like	**bad**	understand	book	**live**	brought	grant
would	instead	often	**care**	**like**	**devil**	past
could	villag	**feel**	connect	**get**	lean	deep
want	**class**	lack	**great**	**walk**	number	statu
peopl	state	claud	win	might	film	understand
know	school	not-claud	colleg	**feel**	**sad**	perhap
go	novel	season	student	**along**	meet	steve
never	chin	baker	generat	**tast**	**expect**	hole

Table 2: Top 10 most probable words for each sentiment. Bold words appear in the prior lexicon. Green words have the same label in the prior lexicon; red words changed labels during training.

SubjLDA on pantslessprogressive.com

NEUTRAL	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE
said	photo	al	000	n't	new	know
peopl	imag	**live**	1	**get**	york	**get**
one	**see**	post	**attack**	**like**	**near**	ita
state	**feel**	video	rep	**look**	water	think
would	much	libya	coast	thing	u	donat
govern	news	abdi	ivori	**see**	**feel**	go
presid	stori	facebook	2	**live**	c	say
year	**hate**	syrian	100	donat	o	**like**
protest	**despit**	updat	bachmann	**give**	**tear**	paul
say	**charg**	twitter	john	**good**	romney	palin

LDA on pantslessprogressive.com

NEUTRAL	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE
said	palin	court	protest	**like**	percent	**get**
one	**see**	abdi	bahrain	n't	job	post
state	sarah	justic	forc	know	rate	one
would	john	u	syria	think	unemploy	tumblr
peopl	alaska	o	kill	**get**	million	reblog
govern	**class**	000	syrian	donat	**top**	help
presid	**interest**	suprem	citi	peopl	worker	video
year	**murder**	thoma	peopl	go	**like**	donat
say	citizen	c	report	want	make	**top**
time	agent	uu	anti-govern	say	earn	**pleas**

Table 3: Top 10 most probable words for pantslessprogressive.com. Bold words appear in the prior lexicon. Green words have the same label in the prior lexicon; red words changed labels during training.

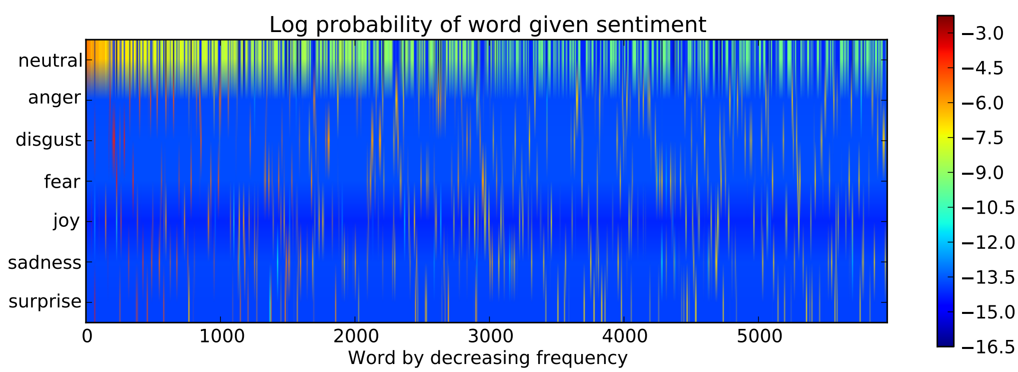

Figure 1: Word mixing proportions in the SubjLDA model for each sentiment for noconsensusonaname.tumblr.com. Words are plotted on the bottom axis, ordered by decreasing frequency. Because most vertical bands have few colors, words are sparsely mixed among topics.

sentiment k, compute its frequency f_k in training corpus. Then, for each word index i, we draw a sentiment label k_i with probability f_{k_i}/V, where V is the size of the corpus vocabulary. Next, draw a word w_i from the sentiment k_i's list with probability g_{w_i}/f_{k_i}. The likelihood of observing any one word is thus

$$\begin{aligned} p(w) &= \sum_{k=1}^{K} p(w|k)p(k) \\ &= \sum_{k=1}^{K} \frac{g_w}{V} \end{aligned} \tag{3}$$

4.3 Derivation of SubjLDA's Likelihood of New Data

As Lin did not derive a conditional distribution for the per-document likelihood of SubjLDA, so we derive it here. Given the above MAP estimates of these model parameters, we simply marginalize over the subjectivity labels $\mathbf{s}$ and sentiment labels $\mathbf{k}$. SubjLDA assumes that given a document, per-sentences subjectivity labels are drawn i.i.d., and given a sentence, per-word sentiment labels are drawn i.i.d.

First, at the end of the Gibbs sampling, we obtain maximum a posteriori estimates of the model parameters $\mathbf{\Pi}$, $\mathbf{\Theta}$, and $\mathbf{\Phi}$ from the samples. This is easy to do because each parameter is a matrix of multinomial proportions, which we estimate by smoothing sample means with our priors. We present Lin's results here [7].

Let $\pi_{d,j}$ denote the probability that sentences in document d takes subjectivity j, $\theta_{d,m,k}$ denote the probability that a word in sentence m of document j takes sentiment k, and $\phi_{k,t}$ denote the probability a word of sentiment k is word t. Then the MAP estimates for our model parameters are

$$\pi_{d,j} = \frac{N_{d,j} + \gamma}{N_d + K\gamma} \tag{4}$$

$$\theta_{d,m,k} = \frac{N_{d,m,k} + \alpha_{s_{d,m},k}}{N_{d,m} + \sum_{j=1}^{S} \alpha_{s_{d,m},k}} \tag{5}$$

$$\phi_{k,t} = \frac{N_{k,t} + \beta_{k,t}}{N_k + \sum_{t=1}^{V} \beta_{k,t}} \tag{6}$$

where $N_{\mathbf{x}}$ denotes the count of all words matching index $\mathbf{x}$, S is the number of sentiments, and V is the vocabulary size.

There is one caveat: $\mathbf{\Theta}$ is a dummy variable, parameterized by a document and sentence counts in the training set. In the likelihood formula for new data, we analytically integrate out Θ and perform our estimations from α, re-estimated by maximum likelihood from the sampling results at the end of training.

Let M denote the number of sentences in a document, N_m denote the number of words in sentence m, and $w_{m,n}$ denote the word at position n in sentence m. Then the likelihood of a new document $\mathbf{w}$ is

$$\begin{aligned} P(\mathbf{w}|\mathbf{\Pi},\mathbf{\Theta},\mathbf{\Phi},\alpha,\beta,\gamma) &= \prod_{m=1}^{M}\sum_{l=1}^{L}\prod_{n=1}^{N}\sum_{k=1}^{K}\int P(w_{m,n}|k,\mathbf{\Phi})P(k|\theta)P(\theta|\alpha_l)d\theta \\ &= \prod_{m=1}^{M}\sum_{l=1}^{L}\prod_{n=1}^{N}\sum_{k=1}^{K}\int \phi_{k,w_{m,n}}\theta_k \frac{\Gamma\left(\sum_{k=1}^{K}\alpha_{l,k}\right)}{\prod_{k=1}^{K}\Gamma(\alpha_{l,k})}\prod_{k=1}^{K}\theta_k^{\alpha_{l,k}-1}d\theta \\ &= \prod_{m=1}^{M}\sum_{l=1}^{L}\prod_{n=1}^{N}\sum_{k=1}^{K}\phi_{k,w_{m,n}} \frac{\Gamma\left(\sum_{k=1}^{K}\alpha_{l,k}\right)}{\prod_{k=1}^{K}\Gamma(\alpha_{l,k})} \frac{\prod_{k'=1}^{K}\Gamma(\alpha_{l,k'}+[k'=k])}{\Gamma\left(\sum_{k'=1}^{K}\alpha_{l,k'}+[k'=k]\right)} \end{aligned} \tag{7}$$

where $[\cdot]$ are Iverson brackets. In the last line, we use the fact that $P(k|\theta)P(\theta|\alpha_l)$ is a Dirichlet distribution with parameter $(\alpha_{l,1},\ldots,\alpha_{l,k}+1,\ldots,\alpha_{l,K})$, so the integral evaluates to 1 and we need only multiply the normalizing constants.

Dataset	Baseline	LDA	SubjLDA
thenatureofdaylight.tumblr.com	219.0	1.573×10^{-3}	5.089×10^{-3}
noconsensusonaname.tumblr.com	747.9	1.317×10^{-3}	2.229×10^{-3}
pantslessprogressive.com	422.5	7.411×10^{-4}	1.660×10^{-3}

Table 4: Perplexity measures for each dataset and each model.

4.4 Derivation LDA's Likelihood of New Data

Since in LDA, θ is also a dummy variable that is valid only for the documents in the training set, we also re-estimate the prior α and then marginalize over θ. The likelihood of a new document $\mathbf{w}$ is

$$\begin{aligned} P(\mathbf{w}|\theta, \mathbf{\Phi}, \alpha, \beta) &= \prod_{n=1}^{N} \sum_{k=1}^{K} \int P(w_n|k, \mathbf{\Phi}) P(k|\theta) p(\theta|\alpha) d\theta \\ &= \prod_{n=1}^{N} \sum_{k=1}^{K} \phi_{k,w_n} \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)} \frac{\prod_{k'=1}^{K} \Gamma(\alpha_{k'} + [k'=k])}{\Gamma\left(\sum_{k'=1}^{K} \alpha_{k'} + [k'=k]\right)} \end{aligned} \tag{8}$$

4.5 Perplexity Results and Discussion

Figure 4 shows that while both LDA and SubjLDA achieve orders of magnitude improvement over the baseline, SubjLDA actually slightly underperforms LDA on all three datasets.

While disappointing from the point of view of SubjLDA, these results simply say that, when applied to in this semi-supervised context, plain LDA marginally outperforms the more complicated (and slower) SubjLDA. In other words, we were unable to replicate Lin's results on our Tumblr dataset, using the 6 emotion categories from WordNet-Affect.

However, the significant improvement of both models over the baseline indicate that solution to evaluating the emotional content of a person's writing already exists: just use LDA.

We end this paper by noting that perplexity, though a standard, well-behaved metric is an unsuitable choice for evaluating emotion models. Perplexity measures the inverse likelihood of an entire sequence of words given a trained model. However, unlike more general problems of topic modeling, we interested only in how closely our model can generate representative *emotional* text rather than evaluate its performance on generating all text. As discussed in Section 2.3, one of our our datasets, fully 83.6% of words in the corpus appeared as neutral words in SentiWordNet. Thus, perplexity unnecessarily penalizes emotion models for potentially inaccurate models of neutral text. This is especially problematic given possibly great asymmetry between neutral and non-neutral terms. But on the other hand, neutrality is significant as it indicates *lack* of emotion, so we cannot simply modify our likelihood functions to ignore the contribution of neutral terms; that would also violate model assumptions by positing that a user never is neutral.

In general, supervised learning is not suitable for personal emotion classification, because at the beginning of inference for a particular user, no labeled text for that user exists. However, an application that learns its user's sentiment online could provide feedback of its inference to the user and improve via reinforcement learning.

Appendix: Post-Mortem

While the results for this project could have been better, I've certainly learned a lot from doing this work.

First, as described in the abstract, *immature* really does capture the state of the art in web sentiment analysis. Several well-cited approaches, including [10, 1, 11] employ elementary techniques. While these discriminative models successfully classify these selected datasets, they give little insight as to the dynamics of a single person's emotional state. Thus, they are of less utility to, say, designing human-computer interfaces to respond according to a user's emotions.

This is partially due to the nature of emotion: all emotions are hidden variables because nobody can read your mind. So any observations, textual, verbal, or physiological are necessarily noise. Words are especially ambiguous at communicating emotion, though they are the most easily accessible data. Nevertheless, good classifications results for large supervised problems have been achieved, a large number of which use SVMs. Sufficient data does indeed overwhelm noise, and does not require much clever trickery to tease out the relationships within the data. This was a lesson often reiterated during class which I am learning by omission. But conversely, if your data is too sparse, then no amount of nonlinear Bayesian trickery may be able to save you.

I had hoped to find some work in the literature building psychological or neurologically-based models for sentiment analysis, but alas these hopes are a bit beyond the time for this field.

Second, I learned just how much of a pain it is to deal with all but the most trivial of datasets. Many results were obtained with a 8.1 million-post Livejournal corpus. I had wanted to compare my my unsupervised approaches with a more typical SVM formulation. Though many papers used this dataset, after hours of crawling the web, I was unable to find a download for myself.

Other datasets were not trivial to acquire or use. Many required filling out a registration form and waiting for days [2]. The data came in a hodge-podge of text formats, requiring lots of crufty parsing code, a distraction from the real data-driven modeling. Scraping your own textual data doesn't give you labels, precluding supervised methods.

References

[1] Ceclilia Ovesdotter Alm, Dan Roth, and Richard Sproat. Emotions from text: machine learning for text-based emotion prediction. In *Proceedings of the Human Language Technology Conference Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, 2005. URL `http://l2r.cs.uiuc.edu/ danr/Papers/AlmRoSp05.pdf`.

[2] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA). ISBN 2-9517408-6-7.

[3] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python*. O'Reilly, Beijing; Cambridge [Mass.], 2009. ISBN 9780596516499 0596516495.

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

[5] Johan Bollen, Huina Mao, and Xiao-Jun Zeng. Twitter mood predicts the stock market. *J. Comput. Science*, 2(1):1–8, 2011.

[6] David Hume. *A Treatise of Human Nature*. 1739.

[7] Chenghua Lin. *Probabilistic topic models for sentiment analysis on the Web*. PhD thesis, University of Exeter, September 2011.

[8] D. M. McNair, M. Lorr, and L. F. Droppleman. Profile of mood states, 1971.

[9] Thomas P. Minka. Estimating a dirichlet distribution, 2000. URL `http://research.microsoft.com/en-us/um/people/minka/papers/dirichlet/`.

[10] Gilad Mishne and Maarten de Rijke. Capturing global mood levels using blog posts. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 145–152. AAAI, 2006.

[11] Sara Owsley Sood and Lucy Vasserman. Esse: Exploring mood on the web esse: Exploring mood on the web. In *Int'l AAAI Conference on Weblogs and Social Media*, 2009.

[12] Carlo Strapparava and Alessandro Valitutti. Wordnet-affect: an affective extension of wordnet. In *In Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 1083–1086, 2004.

[13] Peter D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424. ACL, 2002.