Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph

Michael Speriosu
University of Texas at Austin
speriosu@mail.utexas.edu

Nikita Sudan
University of Texas at Austin
nsudan@utexas.edu

Sid Upadhyay
University of Texas at Austin
sid.upadhyay@utexas.edu

Jason Baldridge
University of Texas at Austin
jbaldrid@mail.utexas.edu

Abstract

There is high demand for automated tools that assign polarity to microblog content such as tweets (Twitter posts), but this is challenging due to the terseness and informality of tweets in addition to the wide variety and rapid evolution of language in Twitter. It is thus impractical to use standard supervised machine learning techniques dependent on annotated training examples. We do without such annotations by using label propagation to incorporate labels from a maximum entropy classifier trained on noisy labels and knowledge about word types encoded in a lexicon, in combination with the Twitter follower graph. Results on polarity classification for several datasets show that our label propagation approach rivals a model supervised with in-domain annotated tweets, and it outperforms the noisily supervised classifier it exploits as well as a lexicon-based polarity ratio classifier.

1 Introduction

Twitter is a microblogging service where users post messages ("tweets") of no more than 140 characters. With around 200 million users generating 140 million tweets per day, Twitter represents one of the largest and most dynamic datasets of user generated content. Along with other social networking websites such as Facebook, the content on Twitter is real time: tweets about everything from a friend's birthday to a devastating earthquake can be found posted during and immediately after an event in question.

This vast stream of real time data has major implications for any entity interested in public opinion and even acting on what is learned and engaging with the public directly. Companies have the opportunity to examine what customers and potential customers are saying about their products and services without costly and time-consuming surveys or explicit requests for feedback. Political organizations and candidates might be able to determine what issues the public is most interested in, as well as where they stand on those issues. Manual inspection of tweets can be useful for many such analyses, but many applications and questions require real-time analysis of massive amounts of social media content. Computational tools that automatically extract and analyze relevant information about opinion expressed on Twitter and other social media sources are thus in high demand.

Full sentiment analysis for a given question or topic requires many stages, including but not limited to: (1) extraction of tweets based on an initial query, (2) filtering out spam and irrelevant items from those tweets, (3) identifying subjective tweets, and (4) identifying the polarity of those tweets. Like most work in sentiment analysis, we focus on the last stage, polarity classification. The simplest approaches are based on the presence of words or emoticons that are indicators of positive or negative polarity (e.g. Twitter's own API, O'Connor et al. (2010)), or calculating a ratio of positive to negative terms (Choi and Cardie, 2009). Though these are a useful first pass, the nuance of language often defeats them (Pang and Lee, 2008). Tweets provide additional challenges compared to edited text; e.g. they are short and include informal/colloquial/abbreviated language.

Proceedings of EMNLP 2011, Conference on Empirical Methods in Natural Language Processing, pages 53–63,
Edinburgh, Scotland, UK, July 27–31, 2011.

Standard supervised classification methods improve the situation somewhat (Pang et al., 2002), but these require texts labeled with polarity as input and they do not adapt to changes in language use. One way around this is to use noisy labels (also referred to as "distant supervision"), e.g. by taking emoticons like ':)' as positive and ':(' as negative, and train a standard classifier (Read, 2005; Go et al., 2009).[1] Semi-supervised methods can also reduce dependence on labeled texts: for example, Sindhwani and Melville (2008) use a polarity lexicon combined with label propagation. Several have used label propagation starting with a small number of hand-labeled words to induce a lexicon for use in polarity classification (Blair-Goldensohn et al., 2008; Rao and Ravichandran, 2009; Brody and Elhadad, 2010).

In this paper, we bring together several of the above approaches via label propagation using modified adsorption (Talukdar and Crammer, 2009). This also allows us to explore the possibility of exploiting the Twitter follower graph to improve polarity classification, under the assumption that people influence one another or have shared affinities about topics. We construct a graph that has users, tweets, word unigrams, word bigrams, hashtags, and emoticons as its nodes; users are connected based on the Twitter follower graph, users are connected to the tweets they created, and tweets are connected to the unigrams, bigrams, hashtags and emoticons they contain. We seed the graph using the polarity values in the OpinionFinder lexicon (Wilson et al., 2005), the known polarity of emoticons, and a maximum entropy classifier trained on 1.8 million tweets with automatically assigned labels based on the presence of positive and negative emoticons, like Read (2005) and Go et al. (2009).

We compare the label propagation approach to the noisily supervised classifier itself and to a standard lexicon-based method using positive/negative ratios. Evaluation is performed on several datasets of tweets that have been annotated for polarity: the Stanford Twitter Sentiment set (Go et al., 2009), tweets from the 2008 debate between Obama and McCain (Shamma et al., 2009), and a new dataset of tweets about health care reform that we have created. In addition to performing standard per-tweet accuracy, we also measure per-target accuracy (for health care reform) and an aggregate error metric over all users in our test set that captures how similar predicted positivity of each user is to their actual positivity. Across all datasets and measures, we find that label propagation is consistently better than the noisily supervised classifier, which in turn outperforms the lexicon-based method. Additionally, for the health care reform dataset, the label propagation approach—which uses no gold labeled tweets, just a hand-created lexicon—outperforms a maximum entropy classifier trained on gold labels. However, we do not find the follower graph to improve performance with our current implementation.

2 Datasets

We use several different Twitter datasets as training or evaluation resources. From the annotated datasets, only tweets with positive or negative polarity are used, so neutral tweets are ignored. While important, subjectivity detection is largely a different problem from polarity classification. For example, Pang and Lee (2004) use minimum cuts in graphs for the former and machine-learned text classification for the latter. We also do not give any special treatment to retweets, though doing so is a possible future improvement.

2.1 Emoticon-based training set (EMOTICON)

Emoticons are commonly exploited as noisy indicators of polarity—including by Twitter's own advanced search "with positive/negative attitude." While imperfect, there is potential for millions of tweets containing emoticons to serve as a source of noisy training material for a supervised classifier. We create such a training set from a sample of the "garden hose"[2] Twitter feed, from September to December, 2009. At the time of collection, this included up to 15% of all tweets worldwide.

From this feed, 6,265,345 tweets containing at least one of the emoticons listed in Table 1 are extracted; 5,156,277 contain a positive emoticon and

[1] Davidov et al. (2010) use 15 emoticons and 50 Twitter hashtags as proxies for sentiment in a similar manner, but their evaluation is indirect. Rather than predicting gold standard sentiment labels, they instead predict whether those same emoticons and hashtags would be appropriate for other tweets.

[2] `http://dev.twitter.com/pages/streaming_api`

+	:) :D =D =) :] =] :-) :-D :-] ;) ;D ;] ;-) ;-D ;-]
−	:(=(:[=[:-(:-[:’(:’[D:

Table 1: Positive and negative emoticons.

+	#ff, congrats, gracias, yay, thx, smile, awesome, hello, excited, moon, loving, glad, sweet, wonderful, birthday, enjoy, goodnight, amazing, cute, bom
−	nickjonas, murphy, brittany, rip, triste, sad, hurts, died, snow, huhu, headache, upset, crying, throat, poor, sucks, ugh, sakit, stomach, horrible

Table 2: Top 20 most predictive common unigram features for the positive and negative classes, in order from more predictive to less predictive.

1,109,068 contain a negative emoticon. A small number of tweets contain both negative and positive emoticons. These are permitted to appear twice, once for each label. Then, a balanced ratio of positive/negative labels is obtained by keeping only 1,109,068 of the positive tweets. Finally, a large proportion of non-English tweets are excluded by a filter that requires a tweet to have at least two words (with at least two characters) from the CMU Pronouncing Dictionary.[3] A few non-English tweets pass through this filter and some English tweets with very unusual words or incorrect spelling are dropped, but this simple strategy works well overall. The final training set contains 1,839,752 tweets, still balanced for positive and negative emoticons.

Table 2 shows the 20 most predictive unigram features of each class in the EmoMaxent classifier (described below) that are among the 1000 most common unigrams in this dataset and are not themselves emoticons. A few non-English (but polarized) words (e.g. *gracias*, *bom*, *triste*) make it past our simple language filter and onto these lists, but the majority of the most predictive words are English. Other highly predictive words are artifacts of the particular tweet sample that comprises the Emoticon dataset, such as ‘nickjonas,’ ‘brittany,’ and ‘murphy,’ the latter two explained by the abundance of negative tweets after actress Brittany Murphy’s death. Most others are intuitively good markers of positive or negative polarity.

Dataset	Use	Size	% Pos
STS	dev	183	59.0
OMD	dev	1898	73.1
HCR-train	train	488	43.2
HCR-dev	dev	534	32.2
HCR-test	test	396	38.6

Table 3: Basic properties of the annotated datasets used in this paper.

2.2 Datasets with polarity annotations

Three annotated datasets, summarized in Table 3 and described below, are used for training, development, or evaluation of polarity classifiers.

Stanford Twitter Sentiment (STS). Go et al. (2009) created a collection of 216 annotated tweets on various topics.[4] Of these, 108 tweets are positive and 75 are negative.

Obama-McCain Debate (OMD). Shamma et al. (2009) used Amazon Mechanical Turk to annotate 3,269 tweets posted during the presidential debate on September 26, 2008 between Barack Obama and John McCain. Each tweet was annotated by one or more Turkers for the categories *positive*, *negative*, *mixed*, or *other*. We filter this dataset with two constraints in order to ensure high inter-annotator agreement. First, at least three votes must have been provided for a tweet to be included. Second, more than half of the votes must have been *positive* or *negative*; the majority label is taken as the gold standard for that tweet. This results in a set of 1,898 tweets. Of these, 705 had positive gold labels and 1192 had negative gold labels, and the average inter-annotator agreement of the Turk votes for these tweets was 83.7%. To our knowledge, we are the first to perform automatic polarity classification on this dataset.

Health Care Reform (HCR). We create a new annotated dataset based on tweets about health care reform in the USA. This was a strongly debated

[3] The dictionary contains 133k English words, including inflected forms and proper nouns. `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

[4] `http://twittersentiment.appspot.com/`

topic that created a large number of polarized tweets, especially in the run up to the signing of the health care bill on March 23, 2010. We extract tweets containing the health care reform hashtag "#hcr" from early 2010; a subset of these are annotated by us and colleagues for polarity (*positive*, *negative*, *neutral*, *irrelevant*) and polarity targets (*health care reform, Obama, Democrats, Republicans, Tea Party, conservatives, liberals,* and *Stupak*). These are separated into training, dev and test sets. As with the other datasets, we restrict attention in this paper only to positive and negative tweets.[5]

2.3 The Twitter follower graph

One of the key ideas we test in this paper is whether social connections can be used to improve polarity classification for individual tweets and users. We construct the Twitter follower graphs for the users in the above datasets in stages using publicly available data from the Twitter API. From the full list of each user's followers, we retain only followers found within the datasets; this prunes unknown users who did not tweet about the topic and thus are unlikely to provide useful information. This method for graph construction offers nearly complete graphs, but has two main disadvantages. First, many users have raised their privacy levels over time, which hinders the ability to view their follower graph. In these cases only their tweet information is known. Secondly, due to the rapid pace of growth on Twitter, user graphs tend to grow quickly; thus our constructed graph is a representation of the user's current social graph and not the exact graph that existed at the time of the tweet.

3 Approach

We compare three main approaches: using lexicon-based positive/negative ratios, maximum entropy classification and label propagation.

3.1 Lexicon-based baseline (LEXRATIO)

A reasonable baseline to use in polarity classification is to count the number of positive and negative terms in a tweet and pick the category with more terms (O'Connor et al., 2010). This actually uses supervision at the level of word types. Like most others, we use the OpinionFinder subjectivity lexicon,[6] which contains 2,304 words annotated as positive and 4,153 words as negative. If the number of positive and negative words in a tweet is equal (including zero for both), the label is chosen at random.

3.2 Maximum entropy classifier (MAXENT)

The OpenNLP Maximum Entropy package[7] is used to train polarity classifiers using either EMOTICON or HCR-TRAIN, henceforth referred to as EMOMAXENT and GOLDMAXENT, respectively. After tokenizing on whitespace, unigram and bigram features are extracted. All characters are lowercased and non-alphanumeric characters are trimmed from the left and right sides of tokens. However, tokens that contain no alphanumeric characters are not trimmed. Stop words[8] are excluded as unigram features. However, bigram features are extracted before stop words are removed since many stop words are informative in the context of content words: e.g., contrast *shit* (negative) from *the shit* (very positive). The beginning and end of tweets are indicated by '$' in bigram features. Thus, the full feature set for the tweet *I love my new iPod Touch! :D* is `[love, ipod, touch, $ i, i love, love my, my ipod, ipod touch, touch :D, :D $]`. The same tokenization method is used for all datasets in this paper.

3.3 Label Propagation (LPROP)

Tweets are not created in isolation—each tweet is linked to other tweets by the same author, and each author is influenced by the tweets of those he or she follows. Common vocabulary and topics of discussion also connect tweets to each other. Graph-based methods such as label propagation (Zhu and Ghahramani, 2002; Baluja et al., 2008; Talukdar and Crammer, 2009) provide a natural means to represent and exploit such relationships in order to improve classification, often while requiring less supervision than with standard classification. Label propagation algorithms spread label distributions from a small set

[5] A public release of this data, along with our code, is available at https://bitbucket.org/speriosu/updown.

[6] http://www.cs.pitt.edu/mpqa/opinionfinderrelease/

[7] http://incubator.apache.org/opennlp/

[8] Taken from: http://www.ranks.nl/resources/stopwords.html

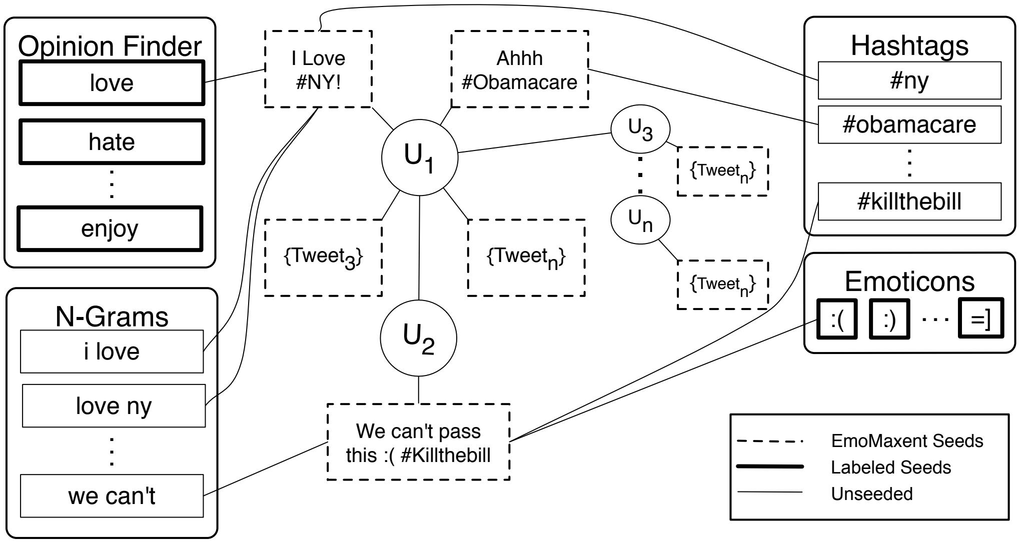

Figure 1: An illustration of our graph with All-edges and Noisy-seed (see text for description).

of nodes seeded with some initial label information (always noisy, heuristic information rather than gold instance labels in our case) throughout the graph. Label distributions are spread across a graph $G = \{V, E, W\}$ where V is the set of n nodes, E is a set of m edges and W is an $n \times n$ matrix of weights, with w_{ij} as the weight of edge (i, j). We use Modified Adsorption (MAD) (Talukdar and Crammer, 2009) over a graph with nodes representing tweets, authors and features, while varying the seed information and the construction of the edge sets. The spreading of the label distributions can be viewed as a controlled random walk with three possible actions: (i) injecting a seeded node with its seed label, (ii) continuing the walk from the current node to a neighboring node, and (iii) abandoning the walk. MAD takes three parameters, μ_1, μ_2 and μ_3, which control the relative importance of each of these actions, respectively. We use the Junto Label Propagation Toolkit's implementation of MAD in this paper.[9]

Modified Adsorption requires some nodes in the graph to have seed distributions, which can come for a variety of knowledge sources. We consider the following variants for seeding the graph:

- **Maxent-seed**: EMOMAXENT is trained on the EMOTICON dataset; every tweet node is seeded with its polarity predictions for the tweet.
- **Lexicon-seed**: Nodes are created for every word in the OpinionFinder lexicon. Positive words are seeded as 90% positive if they are strongly subjective and 80% positive if weakly subjective; similarly and conversely for negative words. Every tweet is connected by an edge to every word in the polarity lexicon it contains, using the weighting scheme discussed with Feature-edges below.
- **Emoticon-seed**: Nodes are created for emoticons from Table 1 and seeded as 90% positive or negative depending on their polarity.
- **Annotated-seed**: The annotations in HCR-TRAIN are used to seed the tweets from that dataset as 100% positive or negative, in accordance with the label.

We use **Noisy-seed** as a collective term for all of the above seed sets except Annotated-seed.

The other main aspect of graph construction is specifying edges and their weights. We consider the following variants:

- **Follower-edges**: When a user A follows another user B, we add an edge from A to B with a weight of 1.0, a weight that is comparable to that of a moderately frequent word in Feature-edges below.
- **Feature-edges**: Nodes are added for hashtags and the features described in §3.2 and connected to the

[9] http://code.google.com/p/junto/

tweets that contain them. An edge connecting a tweet t to a feature f has weight w_{tf} using relative frequency ratios of the feature between the dataset d in question and the EMOTICON dataset as a reference corpus r:

$$w_{tf} = \begin{cases} \log \frac{P_d(f)}{P_r(f)} & if\, P_d(f) > P_r(f) \\ 0 & o.w. \end{cases} \quad (1)$$

We use **All-edges** when combining both edge sets.

Figure 1 illustrates the connections for All-edges and Noisy-seed by example. Each user u_n is attached to anyone who follows them or who they follow. Each user is also connected to the tweets they authored. Words from OpinionFinder are connected to tweets that contain those words, and similarly for hashtags, emoticons, unigrams, and bigrams. Emoticons and words from OpinionFinder are seeded according to the explanation above. All edges other than Feature-edges are given a weight of 1.0.

4 Results

4.1 Parameter tuning

We evaluated our models on the STS, OMD, and HCR-DEV datasets during development and kept HCR-TEST as a final held-out test set used once, after all relevant parameters had been set. For Modified Adsorption, 100 iterations were used, and a seed injection parameter μ_1 of .005 gave the best balance of allowing seed distributions to affect other nodes without overwhelming them. The Junto default value of .01 was used for both μ_2 and μ_3.

4.2 Per-tweet accuracy

Table 4 shows the per-tweet accuracy results of the random baseline, the LEXRATIO baseline, the EMOMAXENT classifier alone, the LPROP classifier run only on Follower-edges with Maxent-seed, the LPROP classifier run on the full graph from Figure 1 only seeded with Lexicon-seed, and the LPROP classifier run on All-edges and Noisy-seed.

For all datasets, LPROP with Feature-edges and Noisy-seed outperforms or matches all other methods. For STS, our best result of 84.7% accuracy beats Go et al. (2009)'s reported best result of 82.7%. Their approach uses a Maxent classifier trained on a noisily labeled emoticon training set similar to our EMOTICON dataset. Note that they also remove neutral tweets from the test set.

Classifier	MSE
Random	.167
LEXRATIO	.170
EMOMAXENT	.233
LPROP (Follower-edges, Maxent-seed)	.233
LPROP (All-edges, Lexicon-seed)	.187
LPROP (Feature-edges, Noisy-seed)	**.148**
LPROP (All-edges, Noisy-seed)	**.148**

Table 5: Mean squared error (MSE) per-user on HCR-TEST, for users with at least 3 tweets

Our semi-supervised label propagation method compares favorably to fully supervised approaches. For example, a graph with Feature-edges seeded with gold labels from HCR-TRAIN (i.e. Annotated-seed) obtains only 64.6% per-tweet accuracy on HCR-TEST. A maximent entropy classifier trained on HCR-TRAIN achieves 66.7%. Our best label propagation approach surpasses both of these at 71.2%.

We find that in general Follower-edges are not helpful as implemented here. Further work is needed to explore more nuanced ways of modeling the social graph, such as allowing leaders to influence followers more than vice versa.

4.3 Per-user error

In many sentiment analysis applications, it is of interest to know what the polarity of a given individual or the overall polarity toward a particular product is. Here we compare the positivity ratio predicted by our methods to that in the gold standard labels on a per-user basis, using the mean squared error between the predicted positivity ratios ppr and the actual ratios apr for all users:

$$MSE(ppr, apr) = \sum_i (apr_i - ppr_i)^2$$

Where apr_i and ppr_i are the actual and predicted positivity ratios of the ith user.

Table 5 gives MSE results on HCR-TEST for users with at least 3 tweets. LPROP (Feature-edges,

Classifier	STS	OMD	HCR-DEV	HCR-TEST
Random	50.0	50.0	50.0	50.0
LexRatio	72.1	59.1	54.3	58.1
EmoMaxent	83.1	61.3	58.6	62.9
LProp (Follower-edges, Maxent-seed)	83.1	61.2	57.9	62.9
LProp (All-edges, Lexicon-seed)	70.0	62.6	64.6	64.6
LProp (Feature-edges, Noisy-seed)	**84.7**	**66.7**	**65.7**	**71.2**
LProp (All-edges, Noisy-seed)	**84.7**	66.5	65.2	71.0

Table 4: Per-tweet accuracy percentages. The models and parameters were developed while tracking performance on STS, OMD, and HCR-DEV, and HCR-TEST results were obtained from a single, blind run.

+	pow pow, good debate, hack the, hack $ barackobama, barackobama, the vp, good job, to vote, john is, is to, obama did, they both, gergen, knowledge, voting for, for veterans, the veterans, america, will take
−	language, this was, drinking, terrorists, government, china, obama i, that we, father, obama in, mc, diplomacy, wars, afghanistan, debt, simply, financial, the spin, the bottom, bottom

Table 7: Top 20 most positive and most negative n-grams in OMD after running LProp with All-edges and Noisy-seed. Note that '$' indicates the beginning or end of a tweet.

+	human, stupak, you do, sunday, fired vote for, yes on, $ we, vote yes, to vote, vote on, goal, nation, do it, up to, ago, votes, this #hcr, #hcr is, on #hcr
−	gop, #tlot #hcr, #tcot #tlot, 12, #topprog, medicare, #tlot, #tlot $, #ocra, cbo, tea party, tea, passes, #hhrs, $ dems, #hc, #obamacare, #sgp, dems, do not

Table 8: Top 20 most positive and most negative n-grams in HCR-TEST after running LProp with All-edges and Noisy-seed.

Noisy-seed) and LProp (All-edges, Noisy-seed) are tied for the lowest error.

4.4 Per-target accuracy

Table 6 gives results on a per-target basis for the five most common targets in the HCR-TEST dataset, in order from most common to least common: *hcr*, *dems*, *obama*, *gop*, and *conservatives*. The percentages reflect the fraction of tweets correctly labeled for each target. These distributions are highly skewed: the *hcr* target covers about 69% of the tweets, while the *conservatives* target covers only about 5%. Thus performance on the *hcr* target tweets is most important for overall accuracy.

5 Discussion

Polar language An attractive property of label propagation algorithms is that label distributions can be obtained for nodes other than the tweets (and importantly, nodes that were unseeded). For example, all of the feature nodes—unigrams, bigrams, and hashtags—have a loading for the positive and negative labels. These could be used for various visualizations of the results of the polarity classification, including terms that are the most positive and negative and also highlighting or bolding such terms when showing a user individual tweets.

Table 7 shows the 20 unigrams and bigrams with the highest and lowest ratio of positive label probability to negative label probability after running LProp with All-edges and Noisy-seed. These lists are restricted to terms that had an edge weight of at least 1.0, i.e. that were twice as frequent in OMD compared to the reference corpus, that had a raw count of at least 5 in OMD, and that didn't already appear in the OpinionFinder lexicon. Some of the terms are intuitively positive and negative, e.g. *good job* and *wars*. Others reflect more specific aspects of the OMD dataset, such as *good debate* and *afghanistan*.

Table 8 shows the top 20 for HCR-TEST. Many

Classifier	hcr (274)	dems (27)	obama (26)	gop (22)	conservatives (20)
LexRatio	58.0	64.8	69.2	50.0	52.5
EmoMaxent	62.4	66.7	73.1	68.2	60.0
LProp (Follower-edges, Maxent-seed)	62.4	66.7	73.1	68.2	60.0
LProp (All-edges, Lexicon-seed)	60.6	85.2	73.1	**86.4**	60.0
LProp (Feature-edges, Noisy-seed)	**69.0**	**81.5**	**80.8**	**86.4**	**70.0**
LProp (All-edges, Noisy-seed)	**69.0**	77.8	**80.8**	**86.4**	**70.0**

Table 6: Per-target accuracy percentages for HCR-test. The number of tweets for each target is given in parentheses.

terms simply reflect a rallying to either pass or defeat the healthcare reform bill (*vote for*, *do not*). Other positive words represent more abstract concepts proponents of the bill may be expressing (*human*, *goal*). Conversely, opponents such as those who would attend a *tea party* are concerned about what they call #*obamacare*.

Domain differences There are several reasons why performance is much lower on both the OMD and HCR datasets than on STS. First, both the Emoticon (noisy) training set and the STS dev set are general in topic. Correct estimations of the positivity and negativity of general words in the training set like *yay* and *upset* are more likely to be useful in a broad-domain evaluation set, whereas misestimations of the weights of more specific words and bigrams are likely to be washed out. In contrast, the OMD and HCR datasets contain a very different vocabulary distribution from the STS set. Words and phrases referring to specific political issues like *health care* and *iraq war* have frequencies that are orders of magnitude higher than either the Emoticon training set or the STS dev set. Thus, misestimations of the positivity or negativity of these features will be amplified in evaluation. Lastly, expression of political opinions tends to be more nuanced than the general opinions and feelings, simply due to the complex nature of political issues. Everyone agrees that a sore throat is bad, while it is less obvious how much government involvement in health care is beneficial.

LexRatio vs. EmoMaxent LexRatio has low coverage for words that tend to indicate positive and negative sentiment in particular domains. For example, STS has the tweet *In montreal for a long weekend of R&R. Much needed*, with a positive gold label. The only word in this tweet in the OpinionFinder lexicon is *long*, which is labeled as negative. Thus, LexRatio incorrectly classifies the tweet as negative. EmoMaxent correctly labels this tweet positive due to features like *weekend* being strong indicators of the positive class. Similarly, the tweet *Booz Allen Hamilton has a bad ass homegrown social collaboration platform. Way cool! #ttiv* is labeled negative by LexRatio due to the presence of *bad*. While EmoMaxent has a negative preference for both *bad* and *ass*, it has a strong positive preference for *bad ass*, as well as both *cool* and *way cool*.

EmoMaxent vs. LProp As seen from the per-tweet and per-user results, LProp does consistently better than Maxent. We now discuss one example of this improvement from the OMD set. One user authored the following four tweets:

- t_1: *obama +3 the conspicuousness of their presence is only matched by our absence #tweetdebate*
- t_2: *Fundamentally, if McCain fundamentally uses "fundamental" one more time, I'm gonna go nuts. #tweetdebate*
- t_3: *McCain likes the bears in Montana joke too much#tweetdebate #current*
- t_4: *We are less respected now... Obama #current #debate08 And I give credit to McCain... NOOO*

The gold label for t_1 is positive and the rest are negative. All of the LProp classifiers correctly predicted the labels for all four tweets. EmoMaxent missed t_2 and t_3, so this primarily negative user is incorrectly indicated as primarily positive by EmoMaxent. LProp gets around this by propagating sentiment polarity through unigram features in this case.

The unigram *mccain* has an edge weight to tweets that contain it of 8.6 for the OMD corpus, meaning *mccain* is much more frequent in this corpus than the reference corpus, so any sentiment associated with *mccain* is propagated strongly. In this case, the output of label propagation seeded with Noisy-seed reveals that *mccain* has negative sentiment for this dataset.

6 Related Work

Much work in sentiment analysis involves the use and generation of dictionaries capturing the sentiment of words. These methods range from manual approaches of developing domain-dependent lexicons (Das and Chan, 2001) to semi-automated approaches (Hu and Liu, 2004) and fully automated approaches (Turney, 2002). Melville et al. (2009) use a unified framework combining background lexical information in terms of word-class associations and refine this information for specific domains using any available training examples. They produce better results than using either a lexicon or training.

O'Connor et al. (2010) use the OpinionFinder subjectivity lexicon to label the polarity of tweets about Barack Obama and compare daily aggregate sentiment scores to the Gallup poll time series of manually gathered approval ratings of Obama. Even with this simple polarity determination, they find significant correlation between their predicted aggregate sentiment per day and the Gallup poll.

Using the OMD dataset, Shamma et al. (2009) find that amount of Twitter activity is a good predictor of topic changes during the debate, and that the content of concurrent tweets reflects a mix of the current debate topic and Twitter users' reactions to that topic. Diakopoulos and Shamma (2010) use the same dataset to develop analysis and visualization techniques to aid journalists and others in understanding the relationship between the live debate event and the timestamped tweets.

Bollen et al. (2010) perform aggregate sentiment analysis on tweets over time, comparing predicted sentiment to time series such as the stock market and crude oil prices, as well as major events such as election day and Thanksgiving. However, the authors use hand-built rules for classification based on the Profile of Mood States (POMS) and largely evaluate based on inspection.

7 Conclusion

We have improved upon existing tweet polarity classification methods by combining several knowledge sources with a noisily supervised label propagation algorithm. We show that a maximum entropy classifier trained with distant supervision works better than a lexicon-based ratio predictor, improving the accuracy for polarity classification on our held-out test set from 58.1% to 62.9%. By using the predictions of that classifier in combination with a graph that incorporates tweets and lexical features, we obtain even better accuracy of 71.2%.

We did not find overall gains from using the follower graph as implemented here. There is room for improvement in the way the follower graph is encoded in our graph, particularly with respect to using asymmetric relationships rather than an undirected graph, and in how follower relationships are weighted.

Another source of information that could be used to improve results is the text in pages that have been linked to from a tweet. In many cases, it is only possible to know what the polarity is by looking at the page being linked to. Our label propagation setup can incorporate this straightforwardly by adding nodes for those pages plus edges between them and all tweets that reference them.

Acknowledgments

This research was supported by a grant from the Morris Memorial Trust Fund of the New York Community Trust. We thank Leif Johnson for providing the tweets from the Twitter firehose for the EMOTICON and HCR datasets, Partha Talukdar for the Junto label propagation toolkit, and the UT Natural Language Learning reading group for helpful feedback.

References

Shumeet Baluja, Rohan Seth, D. Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 895–904, New York, NY, USA, 2008. ACM.

S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. Reis, and J. Reynar. Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era (NLPIX)*, 2008. URL `http://www.ryanmcd.com/papers/local_service_summ.pdf`.

J. Bollen, A. Pepe, and H. Mao. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Proceedings of the 19th International World Wide Web Conference*, 2010.

Samuel Brody and Noemie Elhadad. An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 804–812, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. ISBN 1-932432-65-5. URL `http://portal.acm.org/citation.cfm?id=1857999.1858121`.

Yejin Choi and Claire Cardie. Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 590–598. Association for Computational Linguistics, 2009. URL `http://www.aclweb.org/anthology/D/D09/D09-1062`.

S. Das and M. Chan. Extracting market sentiment from stock message boards. *Asia Pacific Finance Association, 2001*, 2001.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010.

Nicholas A. Diakopoulos and David A. Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the 28th international conference on Human factors in computing systems*, pages 1195–1198, 2010.

Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. Unpublished manuscript. Stanford University, 2009.

Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: http://doi.acm.org/10.1145/1014052.1014073.

Prem Melville, Wojciech Gryc, and Richard D. Lawrence. Sentiment analysis of blogs by combining lexical knowledge with text classification. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1275–1284, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-495-9. doi: http://doi.acm.org/10.1145/1557019.1557156.

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, 2010.

B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86, 2002.

Bo Pang and Lillian Lee. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 2004.

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

Delip Rao and Deepak Ravichandran. Semi-supervised polarity lexicon induction. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 675–682. Association for Computational Linguistics, 2009. URL `http://www.aclweb.org/anthology/E09-1077`.

Jonathon Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, ACLstudent '05, pages 43–48, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. URL `http://portal.acm.org/citation.cfm?id=1628960.1628969`.

David A. Shamma, Lyndon Kennedy, and Elizabeth F. Churchill. Tweet the debates: understanding community annotation of uncollected sources. In *Proceedings of the first SIGMM workshop on Social media*, pages 3–10, 2009.

Vikas Sindhwani and Prem Melville. Document-word co-regularization for semi-supervised sentiment analysis. In *Proceedings of IEEE International Conference on Data Mining (ICDM-08)*, 2008.

Partha Talukdar and Koby Crammer. New regularized algorithms for transductive learning. In Wray Buntine, Marko Grobelnik, Dunja Mladenic, and John Shawe-Taylor, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 5782, pages 442–457. Springer Berlin / Heidelberg, 2009.

P. D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424, 2002.

Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. OpinionFinder: A system for subjectivity analysis. In *Proc. Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP-2005) Companion Volume (software demonstration)*, 2005.

Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.