Cooooooooooooooollllllllllllll!!!!!!!!!!!!!!
Using Word Lengthening to Detect Sentiment in Microblogs
*** Preprint Version ***

Samuel Brody
School of Communication
and Information
Rutgers University
sdbrody@gmail.com

Nicholas Diakopoulos
School of Communication
and Information
Rutgers University
diakop@rutgers.edu

Abstract

We present an automatic method which leverages word lengthening to adapt a sentiment lexicon specifically for Twitter and similar social messaging networks. The contributions of the paper are as follows. First, we call attention to lengthening as a widespread phenomenon in microblogs and social messaging, and demonstrate the importance of handling it correctly. We then show that lengthening is strongly associated with subjectivity and sentiment. Finally, we present an automatic method which leverages this association to detect domain-specific sentiment- and emotion-bearing words. We evaluate our method by comparison to human judgments, and analyze its strengths and weaknesses. Our results are of interest to anyone analyzing sentiment in microblogs and social networks, whether for research or commercial purposes.

1 Introduction

Recently, there has been a surge of interest in sentiment analysis of Twitter messages. Many researchers (e.g., Bollen et al. 2011; Kivran-Swaine and Naaman 2011) are interested in studying structure and interactions in social networks, where sentiment can play an important role. Others use Twitter as a barometer for public mood and opinion in diverse areas such as entertainment, politics and economics. For example, Diakopoulos and Shamma (2010) use Twitter messages posted in conjunction with the live presidential debate between Barack Obama and John McCain to gauge public opinion, Bollen et al. (2010) measure public mood on Twitter and use it to predict upcoming stock market fluctuations, and O'Connor et al. (2010) connect public opinion data from polls to sentiment expressed in Twitter messages along a timeline.

A prerequisite of all such research is an effective method for measuring the sentiment of a post or *tweet*. Due to the extremely informal nature of the medium, and the length restriction[1], the language and jargon which is used in Twitter varies significantly from that of commonly studied text corpora. In addition, Twitter is a quickly evolving domain, and new terms are constantly being introduced. These factors pose difficulties to methods designed for conventional domains, such as news. One solution is to use human annotation. For example, Kivran-Swaine and Naaman (2011) use manual coding of tweets in several emotion categories (e.g., joy, sadness) for their research. Diakopoulos and Shamma (2010) use crowd sourcing via Amazon's Mechanical Turk. Manual encoding usually offers a deeper understanding and correspondingly higher accuracy than shallow automatic methods. However, it is expensive and labor intensive and cannot be applied in real time. Crowd-sourcing carries additional caveats of its own, such as issues of annotator expertise and reliability (see Diakopoulos and Shamma 2010).

The automatic approach to sentiment analysis is commonly used for processing data from social networks and microblogs, where there is often a huge quantity of information and a need for low latency. Many automatic approaches (including all those used in the work mentioned above) have at their core a sentiment lexicon, containing a list of words la-

[1] Messages in Twitter are limited to 140 characters, for compatibility with SMS messaging

beled with specific associated emotions (joy, happiness) or a polarity value (positive, neutral, negative). The overall sentiment of a piece of text is calculated as a function of the labels of the component words. Because Twitter messages are short, shallow approaches are sometimes considered sufficient (Bermingham and Smeaton, 2010). There are also approaches that use deeper machine learning techniques to train sentiment classifiers on examples that have been labeled for sentiment, either manually or automatically, as described above. Recent examples of this approach are Barbosa and Feng (2010) and Pak and Paroubek (2010).

Most established sentiment lexicons (e.g., Wilson et al. 2005, see Section 5) were created for a general domain, and suffer from limited coverage and inaccuracies when applied to the highly informal domain of social networks communication. By creating a sentiment lexicon which is specifically tailored to the microblogging domain, or adapting an existing one, we can expect to achieve higher accuracy and increased coverage. Recent work in this area includes Velikovich et al. (2010), who developed a method for automatically deriving an extensive sentiment lexicon from the web as a whole. The resulting lexicon has greatly increased coverage compared to existing dictionaries and can handle spelling errors and web-specific jargon. Bollen et al. (2010) expand an existing well-vetted psychometric instrument - Profile of Mood States (POMS) (McNair et al., 1971) that associates terms with moods (e.g. calm, happy). The authors use co-occurrence information from the Google n-gram corpus (Brants and Franz, 2006) to enlarge the original list of 72 terms to 964. They use this expanded emotion lexicon (named GPOS) in conjunction with the lexicon of Wilson et al. (2005) to estimate public mood from Twitter posts[2].

The method we present in this paper leverages a phenomenon that is specific to informal social communication to enable the extension of an existing lexicon in a domain specific manner.

[2]Although the authors state that all data and methods will be made available on a public website, it was not present at the time of the writing of this article.

2 Methodology

Prosodic indicators (such as high pitch, prolonged duration, intensity, vowel quality, etc.) have long been know (Bolinger, 1965) as ways for a speaker to emphasize or accent an important word. The ways in which they are used in speech are the subject of ongoing linguistic research (see, for example, Calhoun 2010). In written text, many of these indicators are lost. However, there exist some orthographic conventions which are used to mark or substitute for prosody, including punctuation and typographic styling (italic, bold, and underlined text). In purely text-based domains, such as Twitter, styling is not always available, and is replaced by capitalization or other conventions (e.g., enclosing the word in asterisks). Additionally, the informal nature of the domain leads to an orthographic style which is much closer to the spoken form than in other, more formal, domains. In this work, we hypothesize that the commonly observed phenomenon of lengthening words by repeating letters is a substitute for prosodic emphasis (increased duration or change of pitch). As such, it can be used as an indicator of important words and, in particular, ones that bear strong indication of sentiment.

Our experiments are designed to analyze the phenomenon of lengthening and its implications to sentiment detection. First, in Experiment I, we show the pervasiveness of the phenomenon in our dataset, and measure the potential gains in coverage that can be achieved by considering lengthening when processing Twitter data. Experiment II substantiates the claim that word lengthening is not arbitrary, and is used for emphasis of important words, including those conveying sentiment and emotion. In the first part of Experiment III we demonstrate the implications of this connection for the purpose of sentiment detection using an existing sentiment lexicon. In the second part, we present an unsupervised method for using the lengthening phenomenon to expand an existing sentiment lexicon and tailor it to our domain. We evaluate the method through comparison to human judgments, analyze our results, and demonstrate some of the benefits of our automatic method.

1. For every word in the vocabulary, extract the condensed form, where sequences of a repeated letter are replaced with a single instance of that letter.
   E.g., $niiiice \rightarrow nice$, $realllly \rightarrow realy$ ...
2. Create sets of words sharing the same condensed form.
   E.g., $\{nice, niiice, niccccceee...\}$, $\{realy, really, realllly, ...\}$ ...
3. Remove sets which do not contain at least one repeat of length three.
   E.g.,$\{committee, committe, commitee\}$
4. Find the most frequently occurring form in the group, and mark it as the canonical form.
   E.g., $\{$**nice**$, niiice, niccccceee...\}$, $\{realy,$ **really**$, realllly, ...\}$ ...

Figure 1: Procedure for detecting lengthened words and associating them with a canonical form.

3 Data

Half a million tweets were sampled from the Twitter Streaming API on March 9th 2011. The tweets were sampled to cover a diverse geographic distribution within the U.S. such that regional variation in language use should not bias the data. Some tweets were also sampled from Britain to provide a more diverse sampling of English. We restricted our sample to tweets from accounts which indicated their primary language as English. However, there may be some foreign language messages in our dataset, since multi-lingual users may tweet in other languages even though their account is marked as "English".

The tweets were tokenized and converted to lower-case. Punctuation, as well as links, hashtags, and username mentions were removed. The resulting corpus consists of approximately 6.5 million words, with a vocabulary of 22 thousand words occurring 10 times or more.

4 Experiment I - Detection

To detect and analyze lengthened words, we employ the procedure described in Figure 1. We find sets of words in our data which share a common form and differ only in the number of times each letter is repeated (Steps 1 & 2). In Step 3 we remove sets where all the different forms are likely to be the result of misspelling, rather than lengthening. Finally, in Step 4, we associate all the forms in a single set with a canonical form, which is the most common one observed in the data.

The procedure resulted in 4,359 sets of size > 1. To reduce noise resulting from typos and misspellings, we do not consider words containing non-alphabetic characters, or sets where the canonical form is a single character or occurs less than 10 times. This left us with 3,727 sets.

Analysis Table 1 lists the canonical forms of the 20 largest sets in our list (in terms of the number of variations). Most of the examples are used to express emotion or emphasis. Onomatopoeic words expressing emotion (e.g., *ow, ugh, yay*) are often lengthened and, for some, the combined frequency of the different lengthened forms is actually greater than that of the canonical (single most frequent) one.

Lengthening is a common phenomenon in our dataset. Out of half-a-million tweets, containing roughly 6.5 million words, our procedure identifies 108,762 word occurrences which are lengthenings of a canonical form. These words occur in 87,187 tweets (17.44% or approximately one out of every six, on average). The wide-spread use of lengthening is surprising in light of the length restriction of Twitter. Grinter and Eldridge (2003) point out several conventions that are used in text messages specifically to deal with this restriction. The fact that lengthening is used in spite of the need for brevity suggests that it conveys important information.

Canonical Assumption We validate the assumption that the most frequent form in the set is the canonical form by examining sets containing one or more word forms that were identified in a standard

Can. Form	Card.	# Can.	# Non-Can.
nice	76	3847	348
ugh	75	1912	1057
lmao	70	10085	3727
lmfao	67	2615	1619
ah	61	767	1603
love	59	16360	359
crazy	59	3530	253
yeah	57	4562	373
sheesh	56	247	131
damn	52	5706	299
shit	51	10332	372
really	51	9142	142
oh	51	7114	1617
yay	45	1370	375
wow	45	3767	223
good	45	21042	3171
ow	44	116	499
mad	44	3627	827
hey	44	4669	445
please	43	4014	157

Table 1: The canonical forms of the 20 largest sets (in terms of cardinality), with the number of occurrences of the canonical and non-canonical forms.

English dictionary[3]. This was the case for 2,092 of the sets (56.13%). Of these, in only 55 (2.63%) the most frequent form was *not* recognized by the dictionary. This indicates that the strategy of choosing the most frequent form as the canonical one is reliable and highly accurate ($> 97\%$).

Implications for NLP To examine the effects of lengthening on analyzing Twitter data, we look at the difference in coverage of a standard English dictionary when we explicitly handle lengthened words by mapping them to the canonical form. Coverage with a standard dictionary is important for many NLP applications, such as information retrieval, translation, part-of-speech tagging and parsing. The canonical form for 2,037 word-sets are identified by our dictionary. We searched for occurrences of these words which were lengthened by two or more characters, meaning they would not be identified using standard lemmatization methods or spell-correction techniques that are based on edit distance. We detected 25,101 occurrences of these, appearing in 22,064 (4.4%) tweets. This implies that a lengthening-aware stemming method can be used to increase coverage substantially.

[3]We use the standard dictionary for U.S. English included in the Aspell Unix utility.

5 Experiment II - Relation to Sentiment

At the beginning of Section 2 we presented the hypothesis that lengthening represents a textual substitute for prosodic indicators in speech. As such, it is not used arbitrarily, but rather applied to subjective words to strengthen the sentiment or emotion they convey. The examples presented in Table 1 in the previous section appear to support this hypothesis. In this section we wish to provide experimental evidence for our hypothesis, by demonstrating a significant degree of association between lengthening and subjectivity.

For this purpose we use an existing sentiment lexicon (Wilson et al., 2005), which is commonly used in the literature (see Section 1) and is at the core of OpinionFinder[4], a popular sentiment analysis tool designed to determine opinion in a general domain. The lexicon provides a list of subjective words, each annotated with its degree of subjectivity (strongly subjective, weakly subjective), as well as its sentiment polarity (positive, negative, or neutral). In these experiments, we use the presence of a word (canonical form) in the lexicon as an indicator of subjectivity. It should be noted that the reverse is not true, i.e., the fact that a word is absent from the lexicon does not indicate it is objective.

As a measure of tendency to lengthen a word, we look at the number of distinct forms of that word appearing in our dataset (the cardinality of the set to which it belongs). We group the words according to this statistic, and compare to the vocabulary of our dataset (all words appearing in our data ten times or more, and consisting of two or more alphabetic characters, see Section 4). Figure 2 shows the percentage of subjective words (those in the lexicon) in each of the groups. As noted previously, this is a lower bound, since it is possible (in fact, very likely) that other words in the group are subjective, despite being absent from the lexicon. The graph shows a clear trend - the more lengthening forms a word has,

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

the more likely it is to be subjective (as measured by the percentage of words in the lexicon).

The reverse also holds - if a word is used to convey sentiment, it is more likely to be lengthened. We can verify this by calculating the average number of distinct forms for words in our data that are subjective and comparing to the rest. This calculation yields an average of 2.41 forms for words appearing in our sentiment lexicon (our proxy for subjectivity), compared to an average of 1.79 for those that aren't[5]. This difference is statistically significant at $p < 0.01\%$, using a student t-test.

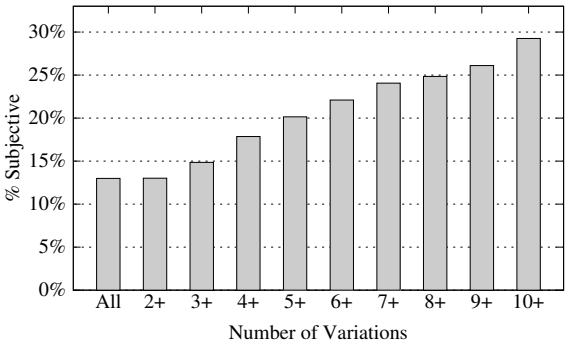
The lexicon we use was designed for a general domain, and suffers from limited coverage (see below) and inaccuracies (see O'Connor et al. 2010 and below Section 6.2 for examples), due to the domain shift. The sentiment lexicon contains 6,878 words, but only 4,939 occur in our data, and only 2,446 appear more than 10 times. Of those appearing in our data, only 485 words (7% of the lexicon vocabulary) are lengthened (the bar for group 2+ in Figure 2), but these are extremely salient. They encompass 701,607 instances (79% of total instances of words from the lexicon), and 339,895 tweets. This provides further evidence that lengthening is used with salient sentiment words.

These results also demonstrates the limitations of using a sentiment lexicon which is not tailored to the domain. Only a small fraction of the lexicon is represented in our data, and it is likely that there are many sentiment words that are commonly used but are absent from it. We address this issue in the next section.

6 Experiment III - Adapting the Sentiment Lexicon

The previous experiment showed the connection between lengthening and sentiment-bearing words. It also demonstrated some of the shortcomings of a lexicon which is not specifically tailored to our domain. There are two steps we can take to use the lengthening phenomenon to adapt an existing sentiment lexicon. The first of these is simply to take lengthening into account when identifying sentiment-bearing words in our corpus. The second is to exploit the connection between lengthening and sentiment to expand the lexicon itself.

[5]This, too, is a conservative estimate, since the later group also includes subjective words, as mentioned.



All	2+	3+	4+	5+	6+	7+	8+	9+	10+
18,817	3,727	2,451	1,540	1,077	778	615	487	406	335

Figure 2: The percentage of subjective word-sets (those whose canonical form appears in the lexicon) as a function of cardinality (number of lengthening variations). The accompanying table provides the total number of sets in each cardinality group.

6.1 Expanding Coverage of Existing Words

We can assess the effect of specifically considering lengthening in our domain by measuring the increase of coverage of the existing sentiment lexicon. Similarly to Experiment I (Section 4), we searched for occurrences of words from the lexicon which were lengthened by two or more characters, and would therefore not be detected using edit-distance. We found 12,105 instances, occurring in 11,439 tweets (2.29% of the total). This increase in coverage is relatively small[6], but comes at almost no cost, by simply considering lengthening in the analysis.

A much greater benefit of lengthening, however, results from using it as an aid in expanding the sentiment lexicon and detecting new sentiment-bearing words. This is the subject of the following section.

6.2 Expanding the Sentiment Vocabulary

In Experiment II (Section 5) we showed that lengthening is strongly associated with sentiment. Therefore, words which are lengthened can provide us with good candidates for inclusion in the lexicon. We can employ existing sentiment-detection meth-

[6]Note that almost half of the increase in coverage calculated in Experiment I (Section 4) comes from subjective words!

ods to decide which candidates to include, and determine their polarity.

Choosing a Candidate Set The first step in expanding the lexicon is to choose a set of candidate words for inclusion. For this purpose we start with words that have 5 or more distinct forms. There are 1,077 of these, of which only 217 (20.15%) are currently in our lexicon (see Figure 2). Since we are looking for commonly lengthened words, we disregard those where the combined frequency of the non-canonical forms is less than 1% that of the canonical one. We also remove stop words, even though some are often lengthened for emphasis (e.g., *me, and, so*), since they are too frequent, and introduce many spurious edges in our co-occurrence graph. Finally, we filter words based on weight, as described below. This leaves us with 720 candidate words.

Graph Approach We examine two methods for sentiment detection - that of Brody and Elhadad (2010) for detecting sentiment in reviews, and that of Velikovich et al. (2010) for finding sentiment terms in a giga-scale web corpus. Both of these employ a graph-based approach, where candidate terms are nodes, and sentiments is propagated from a set of seed words of known sentiment polarity. We calculated the frequency in our corpus of all strongly positive and strongly negative words in the Wilson et al. (2005) lexicon, and chose the 100 most frequent in each category as our seed sets.

Graph Construction Brody and Elhadad (2010) considered all frequent adjectives as candidates and weighted the edge between two adjectives by a function of the number of times they both modified a single noun. Velikovich et al. (2010) constructed a graph where the nodes were 20 million candidate words or phrases, selected using a set of heuristics including frequency and mutual information of word boundaries. Context vectors were constructed for each candidate from all its mentions in a corpus of 4 billion documents, and the edge between two candidates was weighted by the cosine similarity between their context vectors.

Due to the nature of the domain, which is highly informal and unstructured, accurate parsing is difficult. Therefore we cannot employ the exact construction method of Brody and Elhadad (2010). On the other hand, the method of Velikovich et al. (2010) is based on huge amounts of data, and takes advantage of the abundance of contextual information available in full documents, whereas our domain is closer to that of Brody and Elhadad (2010), who dealt with a small number of candidates and short documents typical to online reviews. Therefore, we adapt their construction method. We consider all our candidates words as nodes, along with the words in our positive and negative seed sets. As a proxy for syntactic relationship, edges are weighted as a function of the number of times two words occurred within a three-word window of each other in our dataset. We remove nodes whose neighboring edges have a combined weight of less than 20, meaning they participate in relatively few co-occurrence relations with the other words in the graph.

Algorithm Once the graph is constructed, we can use either of the propagation algorithms of Brody and Elhadad (2010) and Velikovich et al. (2010), which we will denote Reviews and Web, respectively. The Reviews propagation method is based on Zhu and Ghahramani (2002). The words in the positive and negative seed groups are assigned a polarity score of 1 and 0, respectively. All the rest start with a score of 0.5. Then, an update step is repeated. In update iteration t, for each word x that is *not in the seed*, the following update rule is applied:

$$p^t(x) = \frac{\Sigma_{y \in N(x)} w(y,x) \cdot p^{t-1}(y)}{\Sigma_{y \in N(x)} w(y,x)} \qquad (1)$$

Where $p^t(x)$ is the polarity of word x at step t, $N(x)$ is the set of the neighbors of x, and $w(y,x)$ is the weight of the edge connecting x and y. Following Brody and Elhadad (2010), we set this weight to be $1 + log(\#co(y,x))$, where $\#co(y,x)$ is the number of times y and x co-occurred within a three-word window. The update step is repeated to convergence.

Velikovich et al. (2010) employed a different label propagation method, as described in Figure 3. Rather than relying on diffusion along the whole graph, this method considers only the single strongest path between each candidate and each seed word. In their paper, the authors claim that their algorithm is more suitable than that of Zhu and Ghahramani (2002) to a web-based dataset, which

Input: $G = (V, E), w_{ij} \in [0, 1]$
$P, N, \gamma \in \mathbb{R}, T \in \mathbb{N}$
Output: $\text{pol}_i \in \mathbb{R}^{|V|}$
Initialize: $\text{pol}_i, \text{pol}_i^+, \text{pol}_i^- = 0$ for all i
$\text{pol}_i^+ = 1.0$ for all $v_i \in P$ and
$\text{pol}_i^- = 1.0$ for all $v_i \in N$

1: $\alpha_{ij} = 0$ for all $i \neq j$, $\alpha_{ii} = 1$ for all i
2: **for** $v_i \in P$
3: $F = \{v_i\}$
4: **for** $t : 1...T$
5: **for** $(v_k, v_j) \in E$ such that $v_k \in F$
6: $\alpha_{ij} = max(\alpha_{ij}, \alpha_{ik} \cdot w_{k,j})$
$F = F \cup \{v_j\}$
7: **for** $v_j \in V$
8: $\text{pol}_j^+ = \sum_{v_i \in P} \alpha_{ij}$
9: Repeat steps 1-8 using N to compute pol^-
10: $\beta = \sum_i \text{pol}_i^+ / \sum_i \text{pol}_i^-$
11: $\text{pol}_i = \text{pol}_i^+ - \beta \text{pol}_i^-$, for all i
12: if $|\text{pol}_i| < \gamma$ then $\text{pol}_i = 0.0$ for all i

Figure 3: Web algorithm from Velikovich et al. (2010). P and N are the positive and negative seed sets, respectively, w_{ij} are the weights, and T and γ are parameters[9].

contained many dense subgraphs and unreliable associations based only on co-occurrence statistics. We ran both algorithms in our experiment[7], and compared the results.

Evaluation We evaluated the output of the algorithms by comparison to human judgments. For words appearing in the sentiment lexicon, we used the polarity label provided. For the rest, similarly to Brody and Elhadad (2010), we asked volunteers to rate the words on a five-point scale: *strongly-negative, weakly-negative, neutral, weakly-positive,* or *strongly-positive*. We also provided a *N/A* option if the meaning of the word was unknown. Each word was rated by two volunteers. Words which were labeled *N/A* by one or more annotators were considered *unknown*. For the rest, exact inter-rater agreement was 67.6%, but rose to 93% when considering adjacent ratings as equivalent[10]. This is comparable with the agreement reported by Brody and Elhadad (2010). We assigned values 1 (strongly negative) to 5 (strongly positive) to the ratings, and calculated the average between the two ratings for each word. Words with an average rating of 3 were considered neutral, and those with lower and higher ratings were considered negative and positive, respectively.

[7]We normalize the weights described above when using the Web algorithm.

[9]In Velikovich et al. (2010), the parameters T and γ were tuned on a held out dataset. Since our graphs are comparatively small, we do not need to limit the path length T in our search. We do not use the threshold γ, but rather employ a simple cutoff of the top 50 words.

		Human Judgment			
		Pos.	Neg.	Neu.	Unk.
Web	Pos.	18	2	26	2
	Neg.	8	19	17	8
Reviews	Pos.	21	6	21	2
	Neg.	9	14	11	16

Table 2: Evaluation of the top 50 positive and negative words retrieved by the two algorithms through comparison to human judgment.

Web		Reviews	
pos.	neg.	pos.	neg.
see	shit	kidding	rell
win	niggas	justin	whore
way	dis	win	rocks
gotta	gettin	feel	ugg
summer	smh	finale	naw
lets	tight	totally	yea
haha	fuckin	awh	headache
birthday	fuck	boys	whack
tomorrow	sick	pls	yuck
ever	holy	ever	shawty
school	smfh	yer	yeah
peace	outta	lord	sus
soon	odee	mike	sleepy
stuff	wack	three	hunni
canes	nigga	agreed	sick

Table 3: Top fifteen negative and positive words for the algorithms of Brody and Elhadad (2010) (Reviews) and Velikovich et al. (2010) (Web).

Results Table 2 shows the distribution of the human labels among the top 50 most positive and most negative words as determined by the two algorithms. Table 3 lists the top 15 of these as examples.

[10]Cohen's Kappa $\kappa = 0.853$

From Table 2 we can see that both algorithms do better on positive words (fewer words with reversed polarity)[11], and that the Web algorithm is more accurate than the Reviews method. The difference in performance can be explained by the associations used by the algorithms. The Web algorithm takes into account the strongest path to *every* seed word, while the Reviews algorithm propagates from the each seed to its neighbors and then onward. This makes the Reviews algorithm sensitive to strong associations between a word and a single seed. Because our graph is constructed with co-occurrence edges between words, rather than syntactic relations between adjectives, noisy edges are introduced, causing mistaken associations. The Web algorithm, on the other hand, finds words that have a strong association with the positive or negative seed group as a whole, thus making it more robust. This explains some of the examples in Table 3. The words *yeah* and *yea*, which often follow the negative seed word *hell*, are considered negative by the Reviews algorithm. The word *Justin* refers to Justin Beiber, and is closely associated with the positive seed word *love*. Although the Web algorithm is more robust to associations with a single seed, it still misclassifies the word *holy* as negative, presumably because it appears frequently before several different expletives.

Detailed analysis shows that the numbers reported in Table 2 are only rough estimates of performance. For instance, several of the words in the *unknown* category were correctly identified by the algorithm. Examples include *sm(f)h*, which stands for *"shaking my (fucking) head"* and expresses disgust or disdain, *sus*, which is short for *suspicious* (as in *"i hate susssss ass cars that follow me/us when i'm/we walkinggg"*), and *odee*, which means *overdose* and is usually negative (though it does not always refers to drugs, and is sometimes used as an intensifier, e.g., *"aint shit on tv odee bored"*).

There were also cases were the human labels were incorrect in the context of our domain. For example, the word *bull* is listed as positive in the sentiment lexicon, presumably because of its financial sense. In our domain it is (usually) short for *bullshit*. The word *canes* was rated as negative by one of the annotators, but in our data it refers to the Miami Hurricanes, who won a game on the day our dataset was sampled, and were the subject of many positive tweets. This example also demonstrates that our method is capable of detecting terms which are associated with sentiment at different time points, something that is not possible with a fixed lexicon.

[11] This trend is not apparent from the top 15 results presented in Table 3, but becomes noticeable when considering the larger group.

7 Conclusion

In this paper we explored the phenomenon of lengthening words by repeating a single letter. We showed that this is a common phenomenon in Twitter, occurring in one of every six tweets, on average, in our dataset. Correctly detecting these cases is important for comprehensive coverage. We also demonstrated that lengthening is not arbitrary, and is often used with subjective words, presumably to emphasize the sentiment they convey. This finding leads us to develop an unsupervised method based on lengthening for detecting new sentiment bearing words that are not in the existing lexicon, and discovering their polarity. In the rapidly-changing domain of microblogging and net-speak, such a method is essential for up-to-date sentiment detection.

8 Future Work

This paper examined one aspect of the lengthening phenomenon. There are other aspects of lengthening that merit research, such as the connection between the amount of lengthening and the strength of emphasis in individual instances of a word. In addition to sentiment-bearing words, we saw other word classes that were commonly lengthened, including intensifiers (e.g., *very, so, odee*), and named entities associated with sentiment (e.g., *Justin, 'Canes*). These present interesting targets for further study. Also, in this work we focused on data in English, and it would be interesting to examine the phenomenon in other languages. Another direction of research is the connection between lengthening and other orthographic conventions associated with sentiment and emphasis, such as emoticons, punctuation, and capitalization. Finally, we plan to integrate lengthening and its related phenomena into an accurate, Twitter-specific, sentiment classifier.

Acknowledgements

The authors would like to thank Paul Kantor and Mor Naaman for their support and assistance in this project. We would also like to thank Mark Steedman for his help, and the anonymous reviewers for their comments and suggestions.

References

Barbosa, Luciano and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. In Chu-Ren Huang and Dan Jurafsky, editors, *COLING (Posters)*. Chinese Information Processing Society of China, pages 36–44.

Bermingham, Adam and Alan F. Smeaton. 2010. Classifying sentiment in microblogs: is brevity an advantage? In *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, New York, NY, USA, CIKM '10, pages 1833–1836.

Bolinger, Dwight. 1965. *Forms of English: Accent, Morpheme, Order*. Harvard University Press, Cambridge, Massachusetts, USA.

Bollen, J., H. Mao, and X.-J. Zeng. 2010. Twitter mood predicts the stock market. *ArXiv e-prints* .

Bollen, Johan, Bruno Gonalves, Guangchen Ruan, and Huina Mao. 2011. Happiness is assortative in online social networks. *Artificial Life* 0(0):1–15.

Brants, Thorsten and Alex Franz. 2006. Google web 1T 5-gram corpus, version 1. Linguistic Data Consortium, Catalog Number LDC2006T13.

Brody, Samuel and Noemie Elhadad. 2010. 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 (NAACL 2010)*. ACL, Los Angeles, CA, pages 804–812.

Calhoun, Sasha. 2010. The centrality of metrical structure in signaling information structure: A probabilistic perspective. *Language* 86:1–42.

Diakopoulos, Nicholas A. and David A. Shamma. 2010. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the 28th international conference on Human factors in computing systems*. ACM, New York, NY, USA, CHI '10, pages 1195–1198.

Grinter, Rebecca and Margery Eldridge. 2003. Wan2tlk?: everyday text messaging. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, New York, NY, USA, CHI '03, pages 441–448.

Kivran-Swaine, Funda and Mor Naaman. 2011. Network properties and social sharing of emotions in social awareness streams. In *Proceedings of the 2011 ACM Conference on Computer Supported Cooperative Work (CSCW 2011)*. Hangzhou, China.

McNair, D. M., M. Lorr, and L. F. Droppleman. 1971. *Profile of Mood States (POMS)*. Educational and Industrial Testing Service.

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

Pak, Alexander and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. ELRA, Valletta, Malta.

Velikovich, Leonid, Sasha Blair-Goldensohn, Kerry Hannan, and Ryan McDonald. 2010. The viability of web-derived polarity lexicons. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. ACL, Stroudsburg, PA, USA, HLT '10, pages 777–785.

Wilson, Theresa, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. ACL, Stroudsburg, PA, USA, HLT '05, pages 347–354.

Zhu, X. and Z. Ghahramani. 2002. Learning from labeled and unlabeled data with label propagation. Technical report, CMU-CALD-02.