Application of Sentiment Analysis to detect sarcasm in Tweets

Geoffrey Wong

geoffreywong206@gmail.com

Muthanna Chittiappa

kcmuthana@gmail.com

Northeastern University,

Boston, MA

ABSTRACT

Sarcasm is a method of communication that is generally intended to harass or mock someone or a particular situation. Identifying sarcasm through the tone of speech may be easy, but when put in text or rather a tweet, would be harder. Sometimes, it's confused between its literal and intended meaning. Detecting sarcasm in tweets could provide benefits in a variety of areas.
In this paper, we identify four different methods to analyze tweets that are sarcastic. The accuracy of sarcasm among these different methods would be identified. We will devise a hypothesis to determine how sarcasm could possibly be detected in tweets. The hypothesis we propose, despite still in its preliminary version could be used for various applications across social media and other areas of affective computing.

1. INTRODUCTION

First and foremost, let us examine the importance and relevance of sentiment analysis. Sentiment analysis in today's world has huge value in the realm of social media. A lot of organizations use sentiment analysis to determine market strategy, improve their campaign success, improve product messaging, and create systems to reflect sentiment of others, etc. Since the number of people who interact with companies online has been increasing over the years, companies tend to take advantage of this fact and try to determine the end user needs through sentiment analysis. The Oxford dictionary defines sarcasm as, "the use of irony to mock or convey contempt". Sarcasm is not just restricted to the English language despite being used widely. The origins of sarcasm is credited to a Latin word called 'sarkasmos' meaning to gnash the teeth, speak bitterly. Often irony and sarcasm are confused among people. It may be difficult to differentiate between them. Irony is an event or a state of affairs that deliberately contradicts the result of what one expects. The result is usually amusing. The British Broadcasting Corporation (BBC) has an entire webpage dedicated to the sarcasm topic. This page helps non-native English speakers to understand sarcasm in a better way.

In this paper, we are going to address the accuracy of sarcasm detection among tweets from twitter. The data that would be used here is defined in section. Identifying sarcasm in sentences has shown a really low accuracy rate. For instance a study conducted in 2005 showed that only 56% of email statements were detected as sarcastic by participants. Moreover, the participants of that study were confident of conveying the message in statements, although their actual ability to convey the meaning verbally varied significantly. We would like to increase this detection rate further and would explore and formulate methods to increase the detection rate accuracy of sarcasm in the tweets. The benefits of sarcasm can be vast. It particularly has a huge application in natural language processing (NLP) applications. The NLP applications that would benefit from this are dialogue systems, review summarizations to name a few. From a psychological and cognitive perspective, determining the pattern of the occurrences of sarcastic tweets and modeling sarcastic utterances would be an interesting take. Another interesting issue that could be tackled is the ever occurring problem of misinterpretation of tweets from a cyber security perspective. There have been previously recorded instances of violation of law by making sarcastic remarks in tweets. What makes the work unique is data that would be analyzed and the different methodologies that would be implemented to help measure the accuracy rate.

2. RELATED WORK

Determining the accuracy of sarcasm in tweets has been addressed in previous work. Spotter, a well-known French social media analysis company developed a tool for sarcasm detection in tweets. Spotter has claimed that their tools can achieve a detection rate of up to 80%. If this were true we would have actually have a tool that could possibly detect sarcasm better than human beings. But they are yet to provide any concrete proof or evidence to support this. Research papers and documentation of the tool have not been published. A really great related work is the work done by (Davidov, Tsur & Rappoport 2010). In this research paper, the researchers devised a semi-supervised algorithm for sarcasm identification. The data that was used here were tweets that contained the hashtag #sarcasm and Amazon product reviews. The algorithm worked really well with the amazon product reviews mainly because acquiring the gold standard was fairly easy in contrast to the tweets. Tweets that contained sarcasm hashtags seemed noisy. The paper defined tweets as on a scale of 5, with 5 being highly sarcastic and 1 being little or no sarcasm. The highly rated sarcastic tweets were harder for a human to even identify the sarcasm in them. To recognize sarcasm in sentences we can look at the work done by (Wang et al 2006(b); Chen et. al, 2009). The researchers in their study analyzed a variety of speech attributes regarding the vocal tones of sarcastic utterances by users. The attributes that were analyzed were speech rate, mean amplitude, amplitude range among others. Another related work is Liebrecht et. al, (2013). In this paper, the researchers analyzed a dataset of Dutch tweets that used the hashtag #sarcasme. The classifier that was developed for the tweets, determined that most sarcastic tweets contained some sort of a positive message. The classifier of the tweets was developed using the n-gram characteristics to determine sarcasm among the tweets. Another related work is the paper by (Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder 2011). The paper addresses the concern of noise in tweets as mentioned in (Davidov et al 2010). (González-Ibáñez et al 2011) relied on the judgement of tweeters for the separation of the positive and negative sarcastic messages.

This paper also provided a great framework to eliminate the noise.

3. METHODOLOGY

3.1 Acquiring the Dataset

Twitter is a social media site that limits its posting of tweets to 140 characters. So we began our work by first acquiring tweets that had the hashtag sarcasm (#sarcasm) in its context. To do so, we used the API provided by twitter called the streaming API. We also used a library provided by python called Tweepy to write the script to acquire this data. The tweets were acquired in JSON format and had to be converted to csv (excel) format to be more readable. We did so, by writing another script (appendix A). In total, the number of tweets were got over a span of 10 days was 8206. Now with the tweets in csv format we could analyze them.

3.2 SASI Algorithm

The first approach we implemented was the Semi- Supervised Algorithm for Sarcasm Identification. This is the algorithm based on (Davidov et, al, 2009). We planned to use the algorithm as the basis for our work here. In this algorithm, the tweets that are considered for the analysis are classified into different patterns. Now, for instance consider the following tweet: "*A new trend for online retailers? The New J.Crew Catalog Is a Pinterest Page http://ow.ly/oaTap #trnews (@toprank)"*. From this we can identify the user, the url and the hashtag. The dataset would have redundant occurrences of the features. To avoid this redundancy and classify the data we implement these features. Hence the above tweet with tags would look like: *A new trend for online retailers? The New J.Crew Catalog Is a Pinterest Page: [URL], [HASHTAG], [USERNAME]*. For the sarcastic tweets we intend to we use the k-fold cross validation approach to set aside a valid sample. The way we would implement the cross validation would be: k =5. Split the sample into 5 subsets of equal size 2. For each fold estimate a model on all the subsets except one 3. Use the left out subset to test the model, by calculating a CV metric of choice 4. Average the CV metric across subsets to get the CV error. We choose 5 fold cross validation keeping in mind the classification of tweets. We classify the tweets into 5 classes, where class 1 would not be sarcastic tweets and class 5 would be extremely sarcastic tweets. To compare the accuracy rate of the algorithm we would consider tweets that are determined sarcastic by annotators. The difference we did here was the determination of patterns. We classified the tweets into high frequency words (HFW) and content words (CW). HFW words are words that were common such as AND, A, NOT, etc. As for the content words we took the nouns, adverbs, verbs and adjectives. After using this we got a lot of different patterns for each tweet which made it highly difficult to analyze our entire dataset. This is where we failed with implementing the algorithm.

3.3 LIWC2015

LIWC uses a more bag of words approach. Meant as a tool mainly for psychologists to find words in text that are psychology-relevant, it's a simple word counter that sorts them into different tones and parts of speech, and attributing a score based off an internal dictionary. Although this is not as technical as other sentiment analysis techniques, we decided to start with this as this is the standard in the psychology field. We used our licensed LIWC 2015 and ran the dataset across our CSV file.

Looking at the results, we did not achieve what we wished for. Although there were a lot of tweets with multiple emotions, it seems that the posemo and negemo dictionaries (positive and negative emotions) could not overlap. Therefore if a tweet was classified as positive, it could not be negative as well. Therefore, we could not create the contradicting emotions that we wished to achieve.

3.4 NLTK LIBRARY

Based off of a naïve Bayes model, the NLTK sentiment analysis python library seemed to have something useable. In addition to being more accurate, the naïve Bayes model has proven to work for many other text classifications, so it showed promise. Although it only shows positive, negative, and neutral sentiments, it gave us probabilities for both emotions that we could compare. Originally trained on movie review and twitter sentiment, it was comparable to the dataset we were looking at. In addition to the naïve Bayes, they implemented and high information feature selection model. This used a bigram classifier to see if the word was high or low information, and if it was low it is removed. This is very essential as most tweets contain a lot of low information words, and it would decrease the accuracy of the program.

Accessing the API was a simple curl get request, and we wrote a python script to iterate through our data and give the results. For the data, we noticed that although there were a good amount of tweets that had this contrasting emotion, many of the tweets were classified as neutral or in fact had one of the emotions strongly tied to them.

3.5 IBM TONE ANALYSER

We then tried to use another sentiment analysis technique, this time from IBM. It featured an n-gram model, which is a more accurate technique than naïve Bayes and is another method used often in text classification. In addition, the dataset is also trained on punctuation, emoticons, and curse words. Since often the language used in twitter aren't complete words, these additional factors should increase the accuracy.

The API was once again a simple curl get request, and we reused the python script to iterate through the data and get our results. This tone analyzer gives us 5 main emotions: anger, disgust, fear, joy, and sadness. We tried a broader approach, and if any two emotions were similar and greater than 30%, we would consider that a success for our tweets. Although many of our tweets seemed to show promise, overall our core concept of a sarcastic tweet having multiple emotions did not work.

4 Challenges and Assumptions/Dependencies

Proceeding into our evaluation, we had to make some assumptions and were dependent on certain factors. First and foremost, the tweets required needed the hashtag sarcasm in them. This means that we were dependent on the tweeters interpretation of sarcasm. Another dependency for our success was to have tweets that were easily recognizable by humans as sarcastic. Having these assumptions and dependencies helped us in getting tweets that were sarcastic. Acquiring licensing for tools like the LIWC 2015 dictionary was a dependency sorted out by our Professor Stacy Marsella. During our analysis there were some challenges we came across. Most notably were challenges with the dataset. The data was noisy and had to be cleaned. In extracting tweets with the hashtag sarcasm, we not only got tweets but got a lot of irrelevant information. Hence parsing the data to our tools was always going to be a task. Moreover, there were tweets that were in different languages. We had to clean this up. Cleaning a dataset of 8206 tweets was always going to be tedious since we had to manually do so. Going forward automating this process would be a great way to go about it. Not all tweets with the hashtag sarcasm were actually sarcastic. We had to identify such tweets and eliminate them from our dataset.

5 Results

We were able to get results from the three tools that we used to analyze our tweets. The output of the LIWC Dictionary can be seen (appendix C). As you can notice here, there is a lot of categories we couldn't use. The results for our NLTK toolkit can be seen (Appendix D). It shows positivity and negativity in our tweets. As for the IBM tone analyzer, we were not able to show the results, since the API was not free and the results could not be saved. But as you may notice we were not able to parse enough tweets to our system to get a promising and conclusive result. Hence we have failed in our analysis and were not able to determine a proper result on getting the accuracy rate of sarcasm detection in tweets.

6 Future Work

For future work, we plan on increasing the efficiency of our methods. A better implementation of the semi supervised algorithm for sarcastic identification could be used more efficiently and the pattern matching aspect of it could be done differently. Writing good code to identify sarcasm and parsing the data automatically to our system is another aspect we could consider going forward with our research.

Analyzing sarcasm can be applied beyond tweets. The approach can be used on various other social media platforms. The analysis could also be applied to various other sentiments and we could also create systems and models of virtual humans to have a better sense to detect sarcasm in conversations. As we have use methods to detect sarcasm in tweets, sarcasm detection could possibly be easy to detect through the tones of speeches and the context preceding it.

7 Conclusion

In this paper, we have experimented and presented new methods to detect sarcasm in tweets. We've applied a variety of approaches ranging from the SASI algorithm to the IBM Watson Tone analyzer. The method we found appropriate and conclusive was the tone analyzer. The tool provided us with different emotion and social summary of the tweets dataset. Looking at the results, we came to a hypothesis that there is a spike in at least two emotions (anger, disgust, fear, joy and sadness), the tweet tends to indicate sarcasm. We noticed this spike for most of the tweets that we parsed into the system.

8 Acknowledgements

The authors would like to thank Professor Stacy Marsella for his ever helpful guidance throughout the entire project and also in helping get the licenses for tools and guiding us in its usage. The authors would also like to acknowledge the Affective Computing Class of Spring 2016 for the healthy criticism and comments provided during our proposal. This helped us shape and perform our analysis in a better way and provided us with a different perspective in the way we conducted and viewed our work. With this we were able to develop a strategy in proceeding with our work.

9 References

R. L. Brown. 1980. The pragmatics of verbal irony. In R. W. Shuy and A. Snukal, editors, *Language use and the uses of language*, pages 111–127. Georgetown University Press.

Clint Burfoot and Timothy Baldwin. 2009. Automatic satire detection: Are you having a laugh? In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 161–164, Suntec, Singapore, August. Association for Computational Linguistics.

H. Clark and R. Gerrig. 1984. On the pretence theory of irony. *Journal of Experimental Psychology: General*, 113:121–126.

Cristian Danescu-Niculescu-Mizil, Gueorgi Kossinets, Jon Kleinberg, and Lillian Lee. 2009. How opinions are received by online communities: A case study on Amazon.com helpfulness votes. Jun.

D. Davidov and A. Rappoport. 2006. Efficient unsupervised discovery of word categories using symmetric patterns and high frequency words. In *COLING-ACL*.

Macmillan English Dictionary. 2007. *Macmillan English Dictionary*. Macmillan Education, 2 edition.

Raymond W Gibbs and Herbert L. Colston, editors. 2007. *Irony in Language and Thought*. Routledge (Taylor and Francis), New York.

R. W. Gibbs and J. E. O'Brien. 1991. Psychological aspects of irony understanding. *Journal of Pragmatics*, 16:523–530.

R. Gibbs. 1986. On the psycholinguistics of sarcasm. *Journal of Experimental Psychology: General*, 105:3–15.

H. P. Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and semantics*, volume 3. New York: Academic Press.

Minqing Hu and Bing Liu. 2004. 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. ACM.

A. Katz. 2005. Discourse and social-cultural factors in understanding non literal language. In Colston H. and Katz A., editors, *Figurative language comprehension: Social and cultural influences*, pages 183–208. Lawrence Erlbaum Associates.

Jingjing Liu, Yunbo Cao, Chin-Yew Lin, Yalou Huang, and Ming Zhou. 2007. Low-quality product review detection in opinion summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational*

Natural Language Learning (EMNLPCoNLL), pages 334–342.
Rada Mihalcea and Stephen G. Pulman. 2007. Characterizing humour: An exploration of features in humorous texts. In *CICLing*, pages 337–347.
Rada Mihalcea and Carlo Strapparava. 2005. Making computers laugh: Investigations in automatic humor recognition. Pages 531–538, Vancouver, Canada.
D.C. Muecke. 1982. *Irony and the ironic*. Methuen, London, New York.
Bo Pang and Lillian Lee. 2004. A sentimental education:
Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the ACL*, pages 271–278.
Bo Pang and Lillian Lee. 2008. *Opinion Mining and Sentiment Analysis*. Now Publishers Inc, July. Ana-Maria Popescu and Oren Etzioni. 2005. Extracting product features and opinions from reviews. In *HLT '05: Proceedings of the conference on Human*
Language Technology and Empirical Methods in Natural Language Processing, pages 339–346,
Morristown, NJ, USA. Association for Computational Linguistics.
Frank Jr. Stingfellow. 1994. *The Meaning of Irony*.
State University of NY, New York.
J. Tepperman, D. Traum, and S. Narayanan. 2006.
Yeah right: Sarcasm recognition for spoken dialogue systems. In *InterSpeech ICSLP*, Pittsburgh, PA.
Oren Tsur and Ari Rappoport. 2009. Revrank: A fully unsupervised algorithm for selecting the most helpful book reviews. In *International AAAI Conference on Weblogs and Social Media*.
Oren Tsur, Dmitry Davidiv, and Ari Rappoport. 2010.
Icwsm – a great catchy name: Semi-supervised recognition of sarcastic sentences in product reviews.
In *International AAAI Conference on Weblogs and Social Media*.
Akira Utsumi. 1996. A unified theory of irony and its computational formalization. In *COLING*, pages
962–967.
Akira Utsumi. 2000. Verbal irony as implicit display of ironic environment:
Distinguishing ironic utterances from nonirony. *Journal of Pragmatics*, 32(12):1777–1806.
Janyce Wiebe, Theresa Wilson, Rebecca Bruce,
Matthew Bell, and Melanie Martin. 2004. Learning subjective language.
Computational Linguistics,
30(3):277– 308, January.
Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010.
Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, CoNLL 2010.
Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*.
Roger J. Kreuz Gina M. Caucci. 2012. Social and paralinguistic cues to sarcasm. *online 08/02/2012*, 25:1–
22, February.
Roberto Gonz´alez-Ib´a˜nez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.
Lars Kai Hansen, Adam Arvidsson, Finn Arup Nielsen, Elanor Colleoni, and Michael Etter. 2011. Good friends, bad news - affect and virality in twitter. In *The 2011*

International Workshop on Social Computing, Network, and Services (SocialComNet 2011).
Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013.
Tausczik, Yla R., and James W. Pennebaker. "The psychological meaning of words: LIWC and computerized text analysis methods." *Journal of language and social psychology* 29.1 (2010): 24-54.

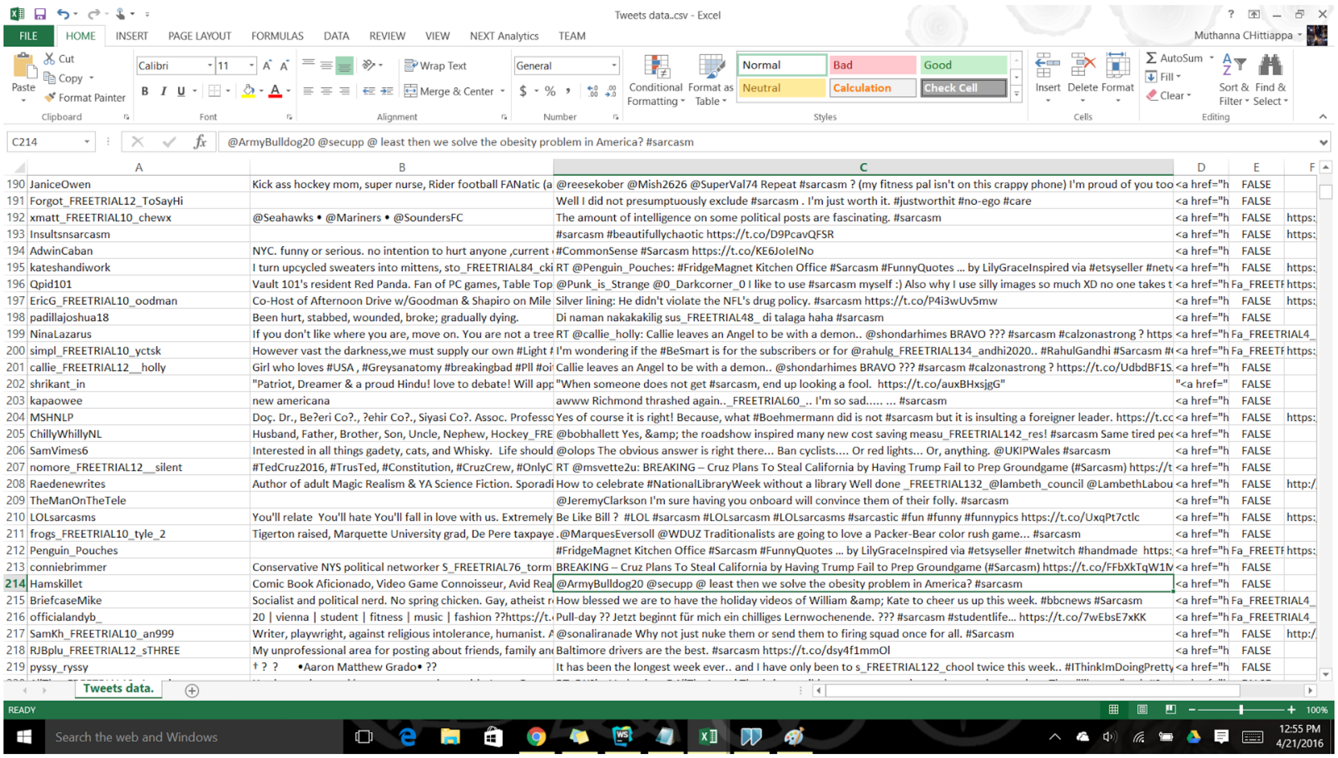
http://streamhacker.com/2010/06/16/text-classification-sentiment-analysis-eliminate-low-information-features/

http://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/tone-analyzer/index.shtml

APPENDIX A: JSON to CSV

```
    )

    args = parser.parse_args()
    input_file = args.input_file
    output_file = args.output_file

    json_data = []
    data = None
    write_header = True
    item_keys = []

    with open(input_file) as json_file:
        json_data = json_file.read()

    try:
        data = json.loads(json_data)
    except Exception as e:
        raise e

    with open(output_file, 'wb') as csv_file:
        writer = csv.writer(csv_file)

        for item in data:
            item_values = []
            for key in item:
                if write_header:
                    item_keys.append(key)

                value = item.get(key, '')
                if isinstance(value, StringTypes):
                    item_values.append(value.encode('utf-8'))
                else:
                    item_values.append(value)

            if write_header:
                writer.writerow(item_keys)
                write_header = False

            writer.writerow(item_values)

if __name__ == "__main__":
    main()
```

APPENDIX B: Datasheet

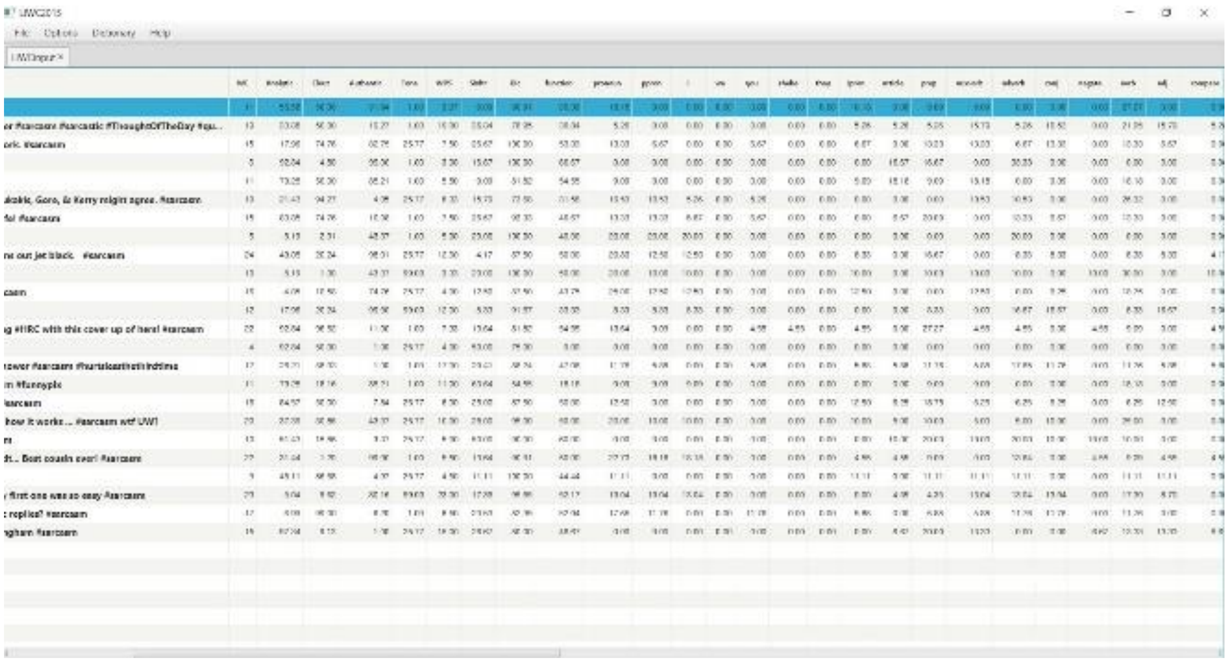


C214 — @ArmyBulldog20 @secupp @ least then we solve the obesity problem in America? #sarcasm

	A	B	C	D	E	F
190	JaniceOwen	Kick ass hockey mom, super nurse, Rider football FANatic (a	@reesekober @Mish2626 @SuperVal74 Repeat #sarcasm ? (my fitness pal isn't on this crappy phone) I'm proud of you too	<a href="h	FALSE	
191	Forgot_FREETRIAL12_ToSayHi		Well I did not presumptuously exclude #sarcasm . I'm just worth it. #justworthit #no-ego #care	<a href="h	FALSE	
192	xmatt_FREETRIAL10_chewx	@Seahawks • @Mariners • @SoundersFC	The amount of intelligence on some political posts are fascinating. #sarcasm	<a href="h	FALSE	https:
193	Insultsnsarcasm		#sarcasm #beautifullychaotic https://t.co/D9PcavQFSR	<a href="h	FALSE	https:
194	AdwinCaban	NYC. funny or serious. no intention to hurt anyone ,current	#CommonSense #Sarcasm https://t.co/KE6JoIeINo	<a href="h	FALSE	
195	kateshandiwork	I turn upcycled sweaters into mittens, sto_FREETRIAL84_cki	RT @Penguin_Pouches: #FridgeMagnet Kitchen Office #Sarcasm #FunnyQuotes ... by LilyGraceInspired via #etsyseller #netv	<a href="h	FALSE	https:
196	Qpid101	Vault 101's resident Red Panda. Fan of PC games, Table Top	@Punk_is_Strange @0_Darkcorner_0 I like to use #sarcasm myself :) Also why I use silly images so much XD no one takes t	<a href="h	Fa_FREETF	https:
197	EricG_FREETRIAL10_oodman	Co-Host of Afternoon Drive w/Goodman & Shapiro on Mile	Silver lining: He didn't violate the NFL's drug policy. #sarcasm https://t.co/P4i3wUv5mw	<a href="h	FALSE	https:
198	padillajoshua18	Been hurt, stabbed, wounded, broke; gradually dying.	Di naman nakakakilig sus_FREETRIAL48_ di talaga haha #sarcasm	<a href="h	FALSE	
199	NinaLazarus	If you don't like where you are, move on. You are not a tree	RT @callie_holly: Callie leaves an Angel to be with a demon.. @shondarhimes BRAVO ??? #sarcasm #calzonastrong ? https	<a href="h	Fa_FREETRIAL4_	
200	simpl_FREETRIAL10_yctsk	However vast the darkness,we must supply our own #Light #	I'm wondering if the #BeSmart is for the subscribers or for @rahulg_FREETRIAL134_andhi2020.. #RahulGandhi #Sarcasm #	<a href="h	Fa_FREETF	https:
201	callie_FREETRIAL12__holly	Girl who loves #USA , #Greysanatomy #breakingbad #Pll #oi	Callie leaves an Angel to be with a demon.. @shondarhimes BRAVO ??? #sarcasm #calzonastrong ? https://t.co/UdbdBF1S.	<a href="h	FALSE	
202	shrikant_in	"Patriot, Dreamer & a proud Hindu! love to debate! Will app	"When someone does not get #sarcasm, end up looking a fool. https://t.co/auxBHxsjgG"	"<a href="	FALSE	
203	kapaowee	new americana	awww Richmond thrashed again.._FREETRIAL60_.. I'm so sad..... ... #sarcasm	<a href="h	FALSE	
204	MSHNLP	Doç. Dr., Be?eri Co?., ?ehir Co?., Siyasi Co?. Assoc. Professo	Yes of course it is right! Because, what #Boehmermann did is not #sarcasm but it is insulting a foreigner leader. https://t.cc	<a href="h	FALSE	https:
205	ChillyWhillyNL	Husband, Father, Brother, Son, Uncle, Nephew, Hockey_FRE	@bobhallett Yes, & the roadshow inspired many new cost saving measu_FREETRIAL142_res! #sarcasm Same tired pe	<a href="h	FALSE	
206	SamVimes6	Interested in all things gadety, cats, and Whisky. Life should	@olops The obvious answer is right there... Ban cyclists.... Or red lights... Or, anything. @UKIPWales #sarcasm	<a href="h	FALSE	
207	nomore_FREETRIAL12__silent	#TedCruz2016, #TrusTed, #Constitution, #CruzCrew, #OnlyC	RT @msvette2u: BREAKING – Cruz Plans To Steal California by Having Trump Fail to Prep Groundgame (#Sarcasm) https://t	<a href="h	FALSE	
208	Raedenewrites	Author of adult Magic Realism & YA Science Fiction. Sporadi	How to celebrate #NationalLibraryWeek without a library Well done _FREETRIAL132_@lambeth_council @LambethLabou	<a href="h	FALSE	http:/
209	TheManOnTheTele		@JeremyClarkson I'm sure having you onboard will convince them of their folly. #sarcasm	<a href="h	FALSE	
210	LOLsarcasms	You'll relate You'll hate You'll fall in love with us. Extremely	Be Like Bill ? #LOL #sarcasm #LOLsarcasm #LOLsarcasms #sarcastic #fun #funny #funnypics https://t.co/UxqPt7ctlc	<a href="h	FALSE	https:
211	frogs_FREETRIAL10_tyle_2	Tigerton raised, Marquette University grad, De Pere taxpaye	.@MarquesEversoll @WDUZ Traditionalists are going to love a Packer-Bear color rush game... #sarcasm	<a href="h	FALSE	
212	Penguin_Pouches		#FridgeMagnet Kitchen Office #Sarcasm #FunnyQuotes ... by LilyGraceInspired via #etsyseller #netwitch #handmade https:	<a href="h	Fa_FREETF	https:
213	conniebrimmer	Conservative NYS political networker S_FREETRIAL76_torm	BREAKING – Cruz Plans To Steal California by Having Trump Fail to Prep Groundgame (#Sarcasm) https://t.co/FFbXkTqW1M	<a href="h	FALSE	
214	Hamskillet	Comic Book Aficionado, Video Game Connoisseur, Avid Rea	@ArmyBulldog20 @secupp @ least then we solve the obesity problem in America? #sarcasm	<a href="h	FALSE	
215	BriefcaseMike	Socialist and political nerd. No spring chicken. Gay, atheist r	How blessed we are to have the holiday videos of William & Kate to cheer us up this week. #bbcnews #Sarcasm	<a href="h	Fa_FREETRIAL4_	
216	officialandyb_	20 | vienna | student | fitness | music | fashion ??https://t.	Pull-day ?? Jetzt beginnt für mich ein chilliges Lernwochenende. ??? #sarcasm #studentlife... https://t.co/7wEbsE7xKK	<a href="h	Fa_FREETRIAL4_	
217	SamKh_FREETRIAL10_an999	Writer, playwright, against religious intolerance, humanist. A	@sonaliranade Why not just nuke them or send them to firing squad once for all. #Sarcasm	<a href="h	FALSE	http:/
218	RJBplu_FREETRIAL12_sTHREE	My unprofessional area for posting about friends, family an	Baltimore drivers are the best. #sarcasm https://t.co/dsy4f1mmOl	<a href="h	FALSE	
219	pyssy_ryssy	† ? ? •Aaron Matthew Grado• ??	It has been the longest week ever.. and I have only been to s_FREETRIAL122_chool twice this week.. #IThinkImDoingPretty	<a href="h	FALSE	

Tweets data.

APPENDIX C: LIWC Output



APPENDIX D: NLTK

```
geo@home-pc ~
$ curl -d "text=If all people without Native American heritage would just go back where they came from wouldn't America b
e great again? " http://text-processing.com/api/sentiment/
{"probability": {"neg": 0.51425411402661259, "neutral": 0.8725925028271363, "pos": 0.48574588597338741}, "label": "neutra
l"}
geo@home-pc ~
```