#SarcasmDetection is soooo general! Towards a Domain-Independent Approach for Detecting Sarcasm

Abstract

Automatic sarcasm detection methods have traditionally been designed for maximum performance on a specific domain. This poses challenges for those wishing to transfer those approaches to other existing or novel domains, which may be typified by very different language characteristics. We develop a general set of features and evaluate it under different training scenarios utilizing in-domain and/or out-of-domain training data. The best-performing scenario, training on both while employing a domain adaptation step, achieves an F1 of 0.780, which is well above baseline F1-measures of 0.515 and 0.345. We also show that the approach outperforms the best results from prior work on the same target domain.

Introduction

Sarcasm, a creative device used to communicate an intended meaning that is actually the opposite of its literal meaning, is notoriously difficult to convey and interpret through text, in part because doing so relies heavily upon shared contextual understandings that can be marked more easily by altered prosody (e.g., emphasis upon certain words) or non-verbal signals (e.g., rolling one's eyes). It is a complex process even for humans, and in fact an inability to detect sarcasm has been linked with a number of neurocognitive disorders, including dementia BIBREF0 . It is similarly a challenging open task in natural language processing, and has direct implications to a number of other critical application areas, such as sentiment analysis.

Most research on automatic sarcasm detection to date has focused on the Twitter domain, which boasts an ample source of publicly-available data, some of which is already self-labeled by users for the presence of sarcasm (e.g., with #sarcasm). However, Twitter is highly informal, space-restricted, and

subject to frequent topic fluctuations from one post to the next due to the ebb and flow of current events—in short, it is not broadly representative of most text domains. Thus, sarcasm detectors trained using features designed for maximum Twitter performance are not necessarily transferable to other domains. Despite this, it is desirable to develop approaches that can harness the more generalizable information present in the abundance of Twitter data.

In this work, we develop a set of domain-independent features for sarcasm detection and show that the features generally perform well across text domains. Further, we validate that domain adaptation can be applied to sarcasm detection to leverage patterns in out-of-domain training data, even when results from training only on that source domain data are extremely bad (far below baseline results), to improve over training on only the target data or over training on the simply combined dataset. Finally, we make a new dataset of sarcastic and non-sarcastic tweets available online as a resource to other researchers.

Related Work

The majority of work on automatic sarcasm detection has been done using Twitter, and to a smaller extent Amazon product reviews. Research outside of those domains has been scarce, but interesting. Notably, Burfoot and Baldwin Burfoot:2009:ASD:1667583.1667633 automatically detected satirical news articles using unigrams, lexical features, and semantic validity features, and Justo et al. Justo2014124 used n-gram, linguistic, and semantic features to detect the presence of sarcasm in the Internet Argument Corpus BIBREF1 . The remainder of this section describes prior work with Twitter and Amazon.

Sarcasm Detection on Twitter

Twitter is a micro-blogging service that allows users to post short “tweets” to share content or describe their feelings or opinions in 140 characters or less. For researchers, it boasts a low cost of annotation and

plentiful supply of data (users often self-label their tweets using the “#” symbol—many explicitly label their sarcastic tweets using the hashtag “#sarcasm”). A variety of approaches have been taken toward automatically detecting sarcasm on Twitter, including explicitly using the information present in a tweet's hashtag(s); Maynard and Greenwood maynard2014cares learned which hashtags characteristically corresponded with sarcastic tweets, and used the presence of those indicators to predict other sarcastic tweets, with high success. BIBREF2 liebrecht2013perfect detected sarcasm in Dutch tweets using unigram, bigram, and trigram features.

BIBREF3 Rajadesingan:2015:SDT:2684822.2685316 detected sarcastic tweets based on features adapted from behavioral models of sarcasm usage, drawing extensively from individual users' Twitter histories and relying heavily on situational context and user characteristics. The system also employed lexical features and grammatical correctness as a means of modelling different aspects of the user's behavior.

Other researchers have had success identifying sarcasm by a tweet's use of positive sentiment to describe a negative situation BIBREF4 , employing contextual BIBREF5 or pragmatic BIBREF6 features, and observing the writing style and emotional scenario of a tweet BIBREF7 . An underlying theme among these methods is that the features are generally designed specifically for use with tweets. A major challenge in developing a more general approach for sarcasm detection lies in developing features that are present across many domains, yet still specific enough to reliably capture the differences between sarcastic and non-sarcastic text.

Finally, some researchers have recently explored approaches that rely on word embeddings and/or carefully tailored neural networks, rather than on task-specific feature design BIBREF8 , BIBREF9 , BIBREF10 . Since neural networks offer little transparency, it is uncertain whether the features learned in these approaches would be easily transferable across text domains for this task (prior research on other

tasks suggests that the features computed by deep neural networks grow increasingly specific to the training dataset—and in turn, to the training domain—with each layer BIBREF11). Although an interesting question, the focus herein is on uncovering the specific types of features capable of leveraging general patterns for sarcasm detection, and this can be more easily examined using shallower learning algorithms.

Sarcasm Detection on Amazon Reviews

Research on automatic sarcasm detection in other domains has been limited, but recently a publicly-available corpus of sarcastic and non-sarcastic Amazon product reviews was released by Filatova FILATOVA12.661 to facilitate research. BIBREF12 buschmeier-cimiano-klinger:2014:W14-26 test many feature combinations on this dataset, including those based on metadata (e.g., Amazon star rating), sentiment, grammar, the presence of interjections (e.g., “wow”) or laughter (e.g., through onomatopoeia or acronyms such as “lol”), the presence of emoticons, and bag-of-words features. Their highest F1 (0.744) is achieved using all of these with a logistic regression classifier; however, using only the star rating, they still achieve an F1 of 0.717. This highlights the need for high-performing, general features for sarcasm detection; metadata features are highly domain-specific, and even bag-of-words trends may be unique to certain domains (“trump” was one of the most common unigrams in our own Twitter training set, but only occurred once across all Amazon product reviews).

Prior to the release of Filatova's dataset, BIBREF13 davidov-tsur-rappoport:2010:CONLL developed a semi-supervised approach to classify tweets or Amazon reviews as sarcastic or non-sarcastic by clustering samples based on grammatical features and the full or partial presence of automatically-extracted text patterns. They evaluated their work on a sample of the classified instances annotated by anonymous users on Amazon Mechanical Turk. They tested several different seed sets with their approach, one of which contained a mixture of positive Amazon reviews, positive #sarcasm-tagged

tweets, and a manually-selected sample of negative tweets. Although they did not report test results on Amazon reviews using this seed set, they did report test results on #sarcasm-tagged tweets, achieving an F-measure of 0.545. Their work is the closest to ours, because it attempts to harness training samples from both the Twitter and Amazon review domains.

Data Collection

Data was taken from two domains: Twitter, and Amazon product reviews. The Amazon reviews were from the publicly available sarcasm corpus developed by Filatova FILATOVA12.661. To build our Twitter dataset, tweets containing exactly one of the trailing hashtags “#sarcasm,” “#happiness,” “#sadness,” “#anger,” “#surprise,” “#fear,” and “#disgust” were downloaded regularly during February and March 2016. Tweets containing the latter six hashtags, corresponding to Ekman's six basic emotions BIBREF14 , were labeled as non-sarcastic. Those hashtags were chosen because their associated tweets were expected to still express opinions, similarly to sarcastic tweets, but in a non-sarcastic way. Tweets containing #sarcasm were labeled as sarcastic; annotating tweets with the #sarcasm hashtag as such is consistent with the vast majority of prior work in the Twitter domain BIBREF6 , BIBREF2 , BIBREF15 , BIBREF3 , BIBREF5 , BIBREF8 , BIBREF10 .

The downloaded tweets were filtered to remove retweets, “@replies,” and tweets containing links. Retweets were removed to avoid having duplicate copies of identical tweets in the dataset, @replies were removed in case the hashtag referred to content in the tweet to which it replied rather than content in the tweet itself, and tweets with links were likewise removed in case the hashtag referred to content in the link rather than in the tweet itself. Requiring that the specified hashtag trailed the rest of the tweet (it could only be followed by other hashtags) was done based on the observation that when sarcastic or emotional hashtags occur in the main tweet body, the tweet generally discusses sarcasm or the specified emotion, rather than actually expressing sarcasm or the specified emotion. Finally, requiring that only one of the

specified hashtags trailed the tweet eliminated cases of ambiguity between sarcastic and non-sarcastic tweets. All trailing “#sarcasm” or emotion hashtags were removed from the data before training and testing, and both datasets were randomly divided into training (80%) and testing (20%) sets. Further details are shown in Table TABREF6 .

Features

Three feature sets were developed (one general, and two targeted toward Twitter and Amazon, respectively). Resources used to develop the features are described in Table TABREF9 . Five classifiers (Naïve Bayes, J48, Bagging, DecisionTable, and SVM), all from the Weka library, were tested using five-fold cross-validation on the training sets, and the highest-scoring (Naïve Bayes) was selected for use on the test set.

The Twitter- (T) and Amazon-specific (A) features are shown in Table TABREF11 . Domain-specific features were still computed for instances from the other domain unless it was impossible to compute those features in that domain (i.e., Amazon Star Rating for Twitter instances), in which case they were left empty. Twitter-specific features are based on the work of BIBREF15 maynard2014cares and BIBREF4 RiloffSarcasm. Maynard and Greenwood detect sarcastic tweets by checking for the presence of learned hashtags that correspond with sarcastic tweets, as well as sarcasm-indicator phrases and emoticons. We construct binary features based on their work, and on Riloff et al.'s work RiloffSarcasm, which determined whether or not a tweet was sarcastic by checking for positive sentiment phrases contrasting with negative situations (both of which were learned from other sarcastic tweets). We also add a feature indicating the presence of laughter terms. Amazon-based features are primarily borrowed from BIBREF12 's buschmeier-cimiano-klinger:2014:W14-26 earlier work on the Amazon dataset. [4]Individual binary features for each of the sarcasm hashtags (5 features) and laughter tokens (9 features) were also included.

We model some of our general features after those from BIBREF4 RiloffSarcasm, under the premise that the underlying principle that sarcasm often associates positive expressions with negative situations holds true across domains. Since positive sentiment phrases and negative situations learned from tweets are unlikely to generalize to different domains, we instead use three sentiment lexicons to build features that capture positive and negative sentiment rather than checking for specific learned phrases. Likewise, rather than bootstrapping specific negative situations from Twitter, we calculate the pointwise mutual information (PMI) between the most positive or negative word in the instance and the n-grams that immediately proceed it to create a more general version of the feature. Other general features developed for this work rely on syntactic characteristics, or are bag-of-words-style features corresponding to the tokens most strongly correlated or most common in sarcastic and non-sarcastic instances from Twitter and Amazon training data. All general features are outlined in Table TABREF14 .

Evaluation

The features used for each train/test scenario are shown in the first column of Table TABREF18 . Twitter Features refers to all features listed in Table TABREF11 preceded by the parenthetical (T), and Amazon Features to all features preceded by (A). General: Other Polarity includes the positive and negative percentages, average polarities, overall polarities, and largest polarity gap features from Table TABREF14 . General: Subjectivity includes the % strongly subjective positive words, % weakly subjective positive words, and their negative counterparts. We also include two baselines: the All Sarcasm case assumes that every instance is sarcastic, and the Random case randomly assigns each instance as sarcastic or non-sarcastic.

Results are reported for models trained only on Twitter, only on Amazon, on both training sets, and on both training sets when Daumé's daumeiii:2007:ACLMain EasyAdapt technique is applied, employing Twitter as the algorithm's source domain and Amazon as its target domain. EasyAdapt works by

modifying the feature space so that it contains three mappings of the original features: a general (source + target) version, a source-only version, and a target-only version. More specifically, assuming an input feature set INLINEFORM0 for some INLINEFORM1 , where INLINEFORM2 is the number of features in the set, EasyAdapt transforms INLINEFORM3 to the augmented set, INLINEFORM4 . The mappings INLINEFORM5 for the source and target domain data, respectively, are defined as INLINEFORM6 and INLINEFORM7 , where INLINEFORM8 is the zero vector. Refer to Daumé daumeiii:2007:ACLMain for an in-depth discussion of this technique.

Each model was tested on the Amazon test data (the model trained only on Twitter was also tested on the Twitter test set). Amazon reviews were selected as the target domain since the Twitter dataset was much larger than the Amazon dataset; this scenario is more consistent with the typically stated goal of domain adaptation (a large labeled out-of-domain source dataset and a small amount of labeled data in the target domain), and most clearly highlights the need for a domain-general approach.
[6]Part-of-speech is considered in MPQA; Amazon and Twitter data was tagged using Stanford CoreNLP BIBREF20 and the Twitter POS-tagger BIBREF21 , respectively.

Finally, we include the best results reported by BIBREF12 buschmeier-cimiano-klinger:2014:W14-26 on the same Amazon dataset. For a more direct comparison between our work and theirs, we also report the results from using all of our features under the same classification conditions as theirs (10-fold cross-validation using scikit-learn's Logistic Regression, tuning with an F1 objective). We refer to the latter case as Our Results, Same Classifier as Prior Best.

Results

The results, including each of the training scenarios noted earlier, are presented in Table TABREF18 . Precision, recall, and F1 on the positive (sarcastic) class were recorded. The highest F1 achieved (0.780)

among all cases was from training on the EasyAdapted Twitter and Amazon data. In comparison, training only on the Amazon reviews produced an F1 of 0.713 (training and testing only on Amazon reviews with our features but with the same classifier and cross-validation settings as BIBREF12 buschmeier-cimiano-klinger:2014:W14-26 led to an F1 of 0.752, outperforming prior best results on that dataset). Training on both without EasyAdapt led to an F1 of 0.595 (or 0.715 when training only on Amazon-specific features), and finally, training only on Twitter data led to an F1 of 0.276. Training and testing on Twitter produced an F1 of 0.583 when training on all features.

Discussion

When testing on Amazon reviews, the worst-performing case was that in which the classifier was trained only on Twitter data (it did not manage to outperform either baseline). This underscores the inherent variations in the data across the two domains; despite the fact that many of the features were deliberately designed to be generalizable and robust to domain-specific idiosyncrasies, the different trends across domains still confused the classifier.

However, combining all of that same Twitter data with a much smaller amount of Amazon data (3998 Twitter training instances relative to 1003 Amazon training instances) and applying EasyAdapt to the combined dataset performed quite well (INLINEFORM0 =0.780). The classifier was able to take advantage of a wealth of additional Twitter samples that had led to terrible performance on their own (INLINEFORM1 =0.276). Thus, the high performance demonstrated when the EasyAdapt algorithm is applied to the training data from the two domains is particularly impressive. It shows that more data is indeed better data—provided that the proper features are selected and the classifier is properly guided in handling it.

Overall, the system cut the error rate from .256 to .220, representing a 14% relative reduction in error

over prior best results on the Amazon dataset. Our results testing on Twitter are not directly comparable to others, since prior work's datasets could not be released; however, our results (INLINEFORM0 =0.583) are in line with those reported previously (BIBREF4 RiloffSarcasm: INLINEFORM1 =0.51; BIBREF13 davidov-tsur-rappoport:2010:CONLL: INLINEFORM2 =0.545). Additionally, our Twitter data did not contain many indicators shown to be discriminative in the past (leading our general features to be better predictors of sarcasm even when training/testing entirely within the domain), and our focus in developing features was on general performance rather than performance on Twitter specifically.

Both datasets were somewhat noisy. Many full-length reviews that were marked as “sarcastic” were only partially so, and included other sentences that were not sarcastic at all. This may have been particularly problematic when strong polarity was present in those sentences. An example of this is shown in Figure FIGREF20 , where the highlighted portion of the review indicates the sarcastic segment submitted by the annotator, and awesome, the most polar word in the entire review (circled), is outside that highlighted sentence.

Since tweets are self-labeled, users' own varying definitions of sarcasm lead to some extreme idiosyncrasies in the kinds of tweets labeled as sarcastic. Sarcastic tweets were also often dependent upon outside context. Some examples include (#sarcasm tags were removed in the actual dataset): “My daughter's 5th grade play went over as professional, flawless, and well rehearsed as a Trump speech. #sarcasm,” “#MilanAlessandria Mario Balotelli scored the fifth goal in the 5-0 win. He should play for the #Azzurri at #EURO2016. #sarcasm,” and “Good morning #sarcasm.”

Finally, some past research has found that it is more difficult to discriminate between sarcastic and non-sarcastic texts when the non-sarcastic texts contain sentiment BIBREF6 , BIBREF8 . Since our non-sarcastic tweets are emotionally-charged, our classifier may have exhibited lower performance than it would have with only neutral non-sarcastic tweets. Since distinguishing between literal and sarcastic

sentiment is useful for real-world applications of sarcasm detection, we consider the presence of sentiment in our dataset to be a worthwhile challenge.

Regarding the general features developed for this work, the polarity- and subjectivity-based features performed well, while performance using only PMI features was lower. PMI scores in particular may have been negatively impacted by common Twitter characteristics, such as the trend to join keywords together in hashtags, and the use of acronyms that are unconventional in other domains. These issues could be addressed to some extent in the future via word segmentation tools, spell-checkers, and acronym expansion.

Conclusions

This work develops a set of domain-independent features and demonstrates their usefulness for general sarcasm detection. Moreover, it shows that by applying a domain adaptation step to the extracted features, even a surplus of “bad” training data can be used to improve the performance of the classifier on target domain data, reducing error by 14% relative to prior work. The Twitter corpus described in this paper is publicly available for research purposes,[2] and represents a substantial contribution to multiple NLP sub-communities. This shared corpus of tweets annotated for sarcasm will hasten the advancement of further research. In the future, we plan to extend our approach to detect sarcasm in a completely novel domain, literature, eventually integrating the work into an application to support reading comprehension.

Acknowledgments

This material is based upon work supported by the NSF Graduate Research Fellowship Program under Grant 1144248, and the NSF under Grant 1262860. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the

views of the National Science Foundation.