Sarcasm Target Identification: Dataset and An Introductory Approach

Aditya Joshi[2,3,4]**, Pranav Goel**[1]**, Pushpak Bhattacharyya**[2]**, Mark J Carman**[3]

[1]IIT (BHU) Varanasi, India, [2]IIT Bombay, India, [3]Monash University, Australia
[4]IITB-Monash Research Academy, India
adityaj@cse.iitb.ac.in, pranav.goel.cse14@iitbhu.ac.in, pb@cse.iitb.ac.in, mark.carman@monash.edu

Abstract

Past work in computational sarcasm deals primarily with sarcasm detection. In this paper, we introduce a novel, related problem: sarcasm target identification (*i.e.*, extracting the target of ridicule in a sarcastic sentence). As a benchmark, we introduce a new dataset for the task. This dataset is manually annotated for the sarcasm target in book snippets and tweets based on our formulation of the task. We then introduce an automatic approach for sarcasm target identification. It is based on a combination of two types of extractors: one based on rules, and another consisting of a statistical classifier. Our introductory approach establishes the viability of sarcasm target identification, and will serve as a baseline for future work.

1. Introduction

Sarcasm is a form of verbal irony that is intended to express contempt or ridicule (Source: The Free Dictionary). While several approaches have been reported for sarcasm detection (Rajadesingan et al., 2015; Joshi et al., 2015; Tsur et al., 2010; González-Ibánez et al., 2011), no past work, to the best of our knowledge, has attempted to identify a crucial component of sarcasm: the target of ridicule (Campbell and Katz, 2012). This is important because the sentiment of the sarcastic text needs to be attributed to this target of ridicule. Towards this motivation, **we introduce 'sarcasm target identification'**: the task of extracting the target of ridicule (*i.e.*, *sarcasm target*) of a sarcastic text. The input is a sarcastic text while the output is either (a) a subset of words in the sentence that point to the sarcasm target, or (b) a fall-back label 'Outside'[1]. In this paper, we **present a manually labeled dataset** consisting of text from two domains: tweets and book snippets. We also **report an automatic approach that takes as input a sarcastic text and returns its sarcasm target**. Our hybrid approach combines a rule-based extractor (that implements a set of rules) and a statistical extractor (that uses a word-level classifier for every word in the sentence, to predict if the word will constitute the sarcasm target).

Since this is the first work in sarcasm target detection, no past work exists to be used as a baseline. Hence, we devise two baselines to validate the strength of our work. The first is a simple, intuitive baseline to show if our approach (which is computationally more intensive than this simple baseline) holds value. In absence of past work, using simple and obvious techniques to solve a problem have been considered as baselines in sentiment analysis (Tan et al., 2011; Pang and Lee, 2005). As the second baseline, we use a technique reported for sentiment/opinion target identification since sentiment target identification appears to be related to sarcasm target identification, on the surface. Our manually labeled datasets are available for download at: `https://github.com/Pranav-Goel/Sarcasm-Target-Detection`. Each unit consists of a piece of text (either book snippet or tweet) with the annotation as the sarcasm target where the sarcasm target is a subset of words in the text or the fall-back label 'Outside'. In addition to this, our hybrid approach for sarcasm target identification will serve as a baseline for future work.

Sarcasm target identification **can benefit natural language generation and sentiment analysis systems**. Being able to recognize the entity towards which the negative sentiment was intended, a natural language generation system will have more context to generate a response. Similarly, a sentiment analysis system will be able to attribute the negative sentiment in a sarcastic text towards the correct aspect of a product or the appropriate entity.

2. Related work

Computational sarcasm primarily focuses on sarcasm detection: classification of a text as sarcastic or non-sarcastic. Joshi et al. (2016a) present a survey of sarcasm detection approaches. They observe three trends in sarcasm detection: semi-supervised extraction of sarcastic patterns, use of hashtag-based supervision, and use of contextual information for sarcasm detection (Tsur et al., 2010; Davidov et al., 2010; Joshi et al., 2015). However, to the best of our knowledge, no past work aims to identify phrases in a sarcastic sentence that indicate the target of ridicule in the sentence.

Related to sarcasm target identification is sentiment target identification. Sentiment target identification deals with identifying the entity towards which sentiment is expressed in a sentence. Qiu et al. (2011) present an approach to extract opinion words and targets collectively from a dataset. Aspect identification for sentiment has also been studied. This deals with extracting aspects of an entity (for example, color, weight, battery in case of a cell phone). Probabilistic topic models have been commonly used for the same. Titov et al. (2008) present a probabilistic topic model that jointly estimates sentiment and aspect in order to achieve sentiment summarization. Lu et al. (2011) perform multi-aspect sentiment analysis using a topic model. Several other topic model-based approaches to aspect extraction have been reported (Mukherjee and Liu, 2012). To the best of our knowledge, ours is the first work that deals with sarcasm target identification.

---

[1]This label is necessary because the sarcasm target may not be present as a word, as discussed in Section 2.

Example	Target
Love when you don't have two minutes to send me a quick text.	you
Don't you just love it when Microsoft tells you that you're spelling your own name wrong.	Microsoft
I love being ignored.	being ignored
He is as good at coding as Tiger Woods is at avoiding controversy.	He, Tiger Woods
Oh, and I suppose the apple ate the cheese.	Outside

Table 1: Examples of sarcasm targets

3. Formulation

Sarcasm is a well-known challenge to sentiment analysis (Pang et al., 2008). Consider the sarcastic sentence '*My cell phone has an awesome battery that lasts 20 minutes*'. This sentence mocks the battery of the cell phone. Aspect-based sentiment analysis deals with identifying sentiment expressed towards different aspects or dimensions of an entity. Therefore, aspect-based sentiment analysis needs to identify that the sentence expresses a negative sentiment towards the aspect '*battery*'. With sarcasm target identification, we hope to enable aspect-based sentiment analysis to attribute the negative sentiment to the correct target aspect in the case of a sarcastic text.

We define the **sarcasm target** as the entity or situation being ridiculed in a sarcastic text. In the case of '*Can't wait to go to class today*', the word '*class*' is the sarcasm target. We make two (fair) assumptions here. (a) Every sarcastic text has at least one sarcasm target. This holds true by definition of sarcasm. Also, (b) The notion of sarcasm target is applicable for sarcastic texts only. A non-sarcastic text does not have a sarcasm target. With these assumptions, we define **sarcasm target identification** as follows. Given a sarcastic text, sarcasm target identification is the task of extracting the subset of words that indicate the target of ridicule. However, in some cases, the target of ridicule may not be present among the words. In such a case, a fall-back label 'Outside' is expected. Examples of some sarcasm targets are given in Table 1.

Some challenges of sarcasm target identification are:

- **Presence of multiple candidate phrases**: Consider the sentence '*This phone heats up so much that I strongly recommend chefs around the world to use it as a cook-top*'. In this sentence, the words '*chefs*', '*cook-top*' and '*phone*' are candidate phrases. However, only the '*phone*' is being ridiculed in this sentence.

- **Multiple sarcasm targets**: A sentence like '*You are as good at coding as he is at cooking*' ridicules both '*you*' and '*he*', and hence, both are sarcasm targets.

- **Absence of a sarcasm target word (the 'Outside' case)**: Consider something bad happens in the beginning of the day and one says, '*What a great way to start off the day!*'. No specific word in the sentence is the sarcasm target. The target here is the situation that started off the day. We refer to such cases as the 'Outside' cases.

4. Dataset

4.1. Collection of sarcastic text

We experiment with two datasets: book snippets and tweets. The dataset of book snippets is a sarcasm-labeled dataset by (Joshi et al., 2016b). From this dataset, 224 book snippets marked as sarcastic are used. The second dataset is dataset of tweets, given by (Riloff et al., 2013). 506 sarcastic tweets from this dataset are used. These book snippets and tweets are manually annotated with the sarcasm target. The statistics of the two datasets are shown in Table 2. The average length of a sarcasm target is 1.6 words in the case of book snippets and 2.08 words in the case of tweets. The last two rows in the table point to an interesting observation. In both the datasets, the average polarity strength[2] of sarcasm target is lower than polarity strength of rest of the sentence. This shows that the sarcasm target is likely to be more neutral than sentiment-bearing.

4.2. Annotation

The annotation is manually carried out by three annotators. The annotators hold more than 5 years of linguistic annotation experience each for sentiment analysis, word sense disambiguation and related tasks. The annotators first discuss the notion of sarcasm target. They are then given the following guiding question for their annotation:
'*The given text is sarcastic. Which words in this text indicate the target of sarcasm that the author making fun of? If you cannot locate specific words, mark them as 'Outside''*.
The target could be an entity or a phrase referring to the situation, and the annotators were told to prefer specific entities over situations when possible.
Every textual unit in both the datasets is labeled for sarcasm target, and the label comprises of either a subset of words in the tweet or the fall-back label 'Outside'. Note that a subset of words means that the word or phrase is taken *exactly as it is* from the sarcastic text being marked for the target - no change in the case, punctuation or wording occurs when the corresponding labels are given. In the case of a phrase, any punctuation at the boundary of the phrase is not included in the sarcasm target/label. In the case of multiple targets, all the targets are annotated.

4.3. Inter-annotator agreement

To understand the challenge posed by sarcasm target annotation, we conduct an additional experiment. Two annotators, different from the original set of expert annotators

---

[2]Polarity strength is the sum of polarities of words. We use a sentiment word-list (McAuley and Leskovec, 2013) to get the strength values

	Snippets	Tweets
Count	224	506
Average #words	28.47	13.06
Vocabulary	1710	1458
Total words	6377	6610
Average length of sarcasm target	1.6	2.08
Average polarity of sarcasm target	0.0087	0.035
Average polarity of portion apart from sarcasm target	0.027	0.53

Table 2: Statistics of our datasets; 'Snippets': Book Snippets

who produced the gold labels, are given the same task of annotation. The annotators are undergraduate students of computer science with English as the primary language of instruction throughout their academic tenure. Following the question as stated above, the new annotators annotate a subset of 50 tweets and 50 book snippets separately from the original annotation and separately from each other.

Since this is a phrase extraction task and not a label assignment task, we use match two proportion-based metrics to quantify the inter-annotator agreement. The first metric is exact match. This metric is defined as the percentage of texts for which the pair of annotations are exactly the same. in the case of the book snippets dataset, the exact match is 28% while in the case of tweets, it is 38%. The higher value in the case of tweets is likely to be due to the more direct nature of ridicule in tweets leading to clearer and more specific targets. The second metric is partial match. This metric is defined as the percentage of texts for which the pair of annotations show a partial overlap. The partial match in the case of both the book snippets and tweets datasets is 66%. There was no overlap in annotations for about one third of the dataset for both book snippets and tweets. The values of exact and partial match hint at the task presenting a reasonable challenge to humans.

5. An Introductory Approach

5.1. Architecture

Our introductory approach for sarcasm target identification is depicted in Figure 1. The input is a sarcastic sentence while the output is the sarcasm target. The approach consists of two kinds of extractors: (a) a **rule-based extractor** that implements nine rules to identify different kinds of sarcasm targets, and (b) a **statistical extractor** that uses statistical classification techniques. The two extractors individually generate lists of candidate sarcasm targets. The third component is the **integrator** that makes an overall prediction of the sarcasm target by choosing among the sarcasm targets returned by the individual extractors. The overall output is a subset of words in the sentence. In case no word is found to be a sarcasm target, a fall-back label 'Outside' is returned. In the forthcoming subsections, we describe the three modules in detail.

5.1.1. Rule-based Extractor

Our rule-based extractor consists of nine rules that take as input the sarcastic sentence, and return a set of candidate
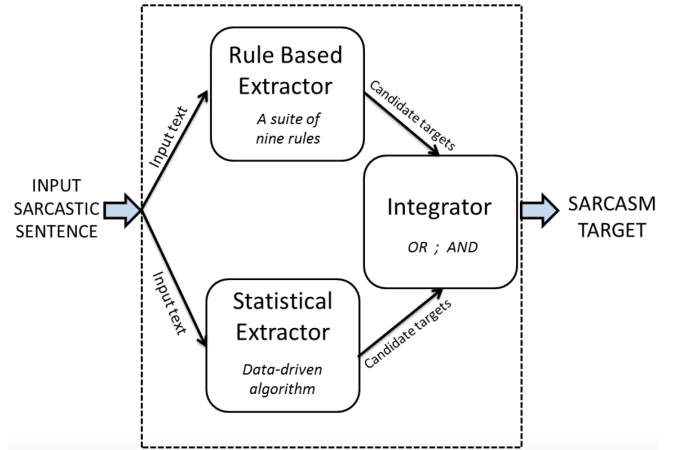

Figure 1: Architecture of our Sarcasm Target Identification Approach

sarcasm targets. The rules are summarized in Table 3. In detail, these rules are as follows:

1. **R1 (Pronouns and Pronominal Adjectives)**: R1 returns pronouns such as '*you, she, they*' and pronominal adjectives (followed by their object) (as in the case of '*your shoes*'). Thus, for the sentence '*I am so in love with my job*', the phrases '*I*' (pronoun) and '*my job*' (based on the pronominal adjective '*my*') are returned as candidate sarcasm targets. This is based on observations by Shamay et al. (2005).

2. **R2 (Named Entities)**: Named entities in a sentence may be sarcasm targets. This rule returns all *named entities* in the sentence. In case of '*Olly Riley is so original with his tweets*', R2 predicts the phrase '*Olly Riley*' as a candidate sarcasm target.

3. **R3 (Sentiment-bearing verb as the pivot)**: This rule is based on the idea by (Riloff et al., 2013) that sarcasm may be expressed as a contrast between a positive sentiment verb and a negative situation. In case of '*I love being ignored*', the sentiment-bearing verb '*love*' is positive. The object of '*love*' is '*being ignored*'. Therefore, R3 returns '*being ignored*' as the candidate sarcasm target. If the sentiment-bearing verb is negative, the rule returns '*Outside*' as a candidate sarcasm target.

4. **R4 (Non-sentiment-bearing verb as the pivot)**: This rule applies in case of sentences where the verb does not bear sentiment. The rule identifies which out of subject or object has a lower sentiment score, and returns the corresponding portion as the candidate sarcasm target. For example, rule R4 returns '*to have a test on my birthday*' as the candidate sarcasm target in case of '*Excited that the teacher has decided to have a test on my birthday!*' where '*decided*' is the non-sentiment-bearing verb. This is also based on Riloff et al. (2013).

5. **R5 (Gerund phrases and Infinitives)**: R5 returns the gerund phrase '*being covered in rashes*' in case of '*Being covered in rashes is fun.*' as the candidate sarcasm

Rule	Definition	Example
R1	Return pronouns and pronominal adjectives	Love when **you** don't have two minutes to send me a quick text .. ; I am so in love with **my job**.
R2	Return named entities as target	Don't you just love it when **Microsoft** tells you that you're spelling your own name wrong.
R3	Return direct object of a positive sentiment verb	I love **being ignored**.
R4	Return phrase on lower sentiment side of primary verb	So happy to just find out it has been *decided* **to reschedule all my lectures and tutorials for me to night classes at the exact same times**!
R5	Return Gerund and Infinitive verb phrases	**Being covered in hives** is so much fun!
R6	Return nouns preceded by a positive sentiment adjective	Yep, this is indeed an *amazing* **donut** ..
R7	Return subject of interrogative sentences	A murderer is stalking me. Could **life** be more fun?
R8	Return subjects of comparisons (similes)	**He** is as good at coding as **Tiger Woods** is at controversies.
R9	Return demonstrative adjective-noun pairs	Oh, I love **this jacket**!

Table 3: Summary of rules in the rule-based extractor; Boldfaced phrases indicate sarcasm targets

target. Similarly, in case of '*Can't wait to wake up early to babysit!*', the infinitive '*to wake up early to babysit*' is returned.

6. **R6 (Noun phrases containing positive adjective):** R6 extracts noun phrases of the form 'JJ NN' where JJ is a positive adjective, and returns the noun indicated by NN. Specifically, 1-3 words preceding the nouns in the sentence are checked for positive sentiment. In case of '*Look at the most realistic walls in a video game*', the noun '*walls*' is returned as the sarcasm target.

7. **R7 (Interrogative sentences):** R7 returns the subject of an interrogative sentence as the sarcasm target. Thus, for '*A murderer is stalking me. Could life be more fun?*', the rule returns '*life*' as the target.

8. **R8 (Sarcasm in Similes):** This rule captures the *subjects/noun phrases* involved in similes and 'as if' comparisons. The rule returns the subject on both sides, as in '*He is as good at coding as Tiger Woods is at avoiding controversy.*' Both '*He*' and '*Tiger Woods*' are returned as targets. This is derived from work on sarcastic similes by Veale et al. (2010).

9. **R9 (Demonstrative adjectives):** This rule captures nouns associated with demonstrative adjectives - *this/that/these/those*. For example, for the sentence '*Oh, I love this jacket!*', R9 returns '*this jacket*' as the sarcasm target.

Combining the outputs of individual rules to generate candidate sarcasm targets of the rule-based extractor: To generate the set of candidate sarcasm targets returned by the rule-based extractor, a weighted majority approach is used as follows. Every rule above is applied to the input sarcastic sentence. Then, every word is assigned a score that sums the accuracy of rules which predicted that this word is a part of the sarcasm target. This accuracy is the overall accuracy of the rule as determined by solely the rule-based classifier. Thus, the integrator weights each word on the basis of how good a rule predicting it as a target was. Words corresponding to the maximum value of this score are returned as candidate sarcasm targets.

5.1.2. Statistical Extractor

The statistical extractor uses a classifier that takes as input a word (along with its features) and returns if the word is a sarcasm target. To do this, we decompose the task into n classification tasks, where n is the total number of words in the sentence. This means that every word in input text is considered as an instance, such that the label can be 1 or 0 depending on whether or not the given word is a part of sarcasm target. For example, '*Tooth-ache is fun*' with sarcasm target as '*tooth-ache*' is broken down into three instances: '*tooth-ache*' with label 1, '*is*' with label 0 and '*fun*' with label 0. In case the target lies outside the sentence, all words have the label 0.

We then represent the instance (*i.e.*, the word) as a set of following features: (A) **Lexical**: *Unigrams*, (B) **Part of Speech (POS)**-based features: *Current POS*, *Previous POS*, *Next POS*, (C) **Polarity**-based features: *Word Polarity* : Sentiment score of the word, *Phrase Polarity* : Sentiment score for the trigram formed by considering the previous word, current word and the next word together (in that order). These polarities lie in the range [-1,+1]. These features are based on our analysis that the target phrase or word tends to be more neutral than the rest of the sentence, and (D) **Pragmatic features**: *Capitalization* : Number of capital letters in the word. Capitalization features are chosen based on features from (Davidov et al., 2010).

The classifiers are trained with words as instances while the sarcasm target is to be computed at the sentence level. Hence, the candidate sarcasm target returned by the statistical extractor consists of words for which the classifier returned 1. For example, the sentence 'He is nice' is broken up into three instances: 'He', 'is' and 'nice'. If the classifier returns 1, 0, 0 for the three instances respectively, the statistical extractor returns 'He' as the candidate sarcasm target. Similarly, if the classifier returns 0, 0, 0 for the three instances, the extractor returns the fall-back label 'Outside'.

5.1.3. Integrator

The integrator determines the sarcasm target based on the outputs of the two extractors. We consider two configurations of the integrator:

1. Hybrid **OR**: In this configuration, the integrator predicts the set of words that occur in the output of either

of the two extractors as the sarcasm target. If the lists are empty, the output is returned as 'Outside'.

2. Hybrid **AND** : In this configuration, the integrator predicts the set of words that occur in the output of both the two extractors as the sarcasm target. If the intersection of the lists is empty, the output is returned as 'Outside'.

The idea of using two configurations OR and AND is based on a rule-based sarcasm detector by (Khattri et al., 2015). While AND is intuitive, the second configuration OR is necessary because our extractors individually may not capture all forms of sarcasm target. This is intuitive because *our rules may not cover all forms of sarcasm targets.*

5.2. Experiment Setup

We use **SVM Perf** (Joachims, 2006) to train the classifiers, optimized for F-score with epsilon e=0.5 and RBF kernel[3]. We set C=1000 for tweets and C=1500 for snippets. We report our results on **four-fold cross validation** for both datasets. <u>Note</u> that we convert individual sentences into words. Therefore, the dataset in the case of book snippets has 6377 instances, while the one of tweets has 6610 instances. The four folds for cross-validation are created over these instances. With a word as instance, the task is binary classification: 1 indicating that the word is a sarcasm target and 0 indicating that it is not. For rules in the rule-based extractor, we use tools in NLTK (Bird, 2006), wherever necessary.

We consider two baselines with which our hybrid approach is compared:

1. **Baseline 1: All Objective Words**: As the first baseline, we design a naïve approach for our task: include all words of the sentence which are not stop words, and have neutral sentiment polarity, as the predicted sarcasm target.

2. **Baseline 2**: Sequence labeling has been reported for opinion target identification (Jin et al., 2009). Therefore, we use **SVM-HMM** (Altun et al., 2003) with default parameters as the second baseline.

We report performance using two metrics: Exact Match Accuracy and Dice Score. These metrics have been used in past work in information extraction (Michelson and Knoblock, 2007). As per their conventional use, these metrics are computed at the sentence level. The metrics are described as:

- **Exact Match (EM) Accuracy**: An exact match occurs if the list of predicted target(s) is exactly the same as the list of actual target(s). The accuracy is computed as number of instances with exact match divided by total instances.

- **Dice Score (DC)**: Dice score(Sørensen, 1948) is used to compare similarity between two samples. This is considered to be a better metric than Exact match accuracy because it accounts for missing words and

_____
[3]RBF Kernel performed better than linear kernel.

extra words in the target. Let the two lists (predicted and actual) be X and Y. Dice score is given by $(2X \cap Y)/(X + Y)$.

5.3. Results

Rule	Overall		Conditional	
	EM	DS	EM	DS
R1	7.14	**32.8**	7.65	35.23
R2	**8.48**	16.7	19.19	37.81
R3	4.91	6.27	16.92	21.62
R4	2.67	11.89	4.38	19.45
R5	1.34	6.39	2.32	11.11
R6	4.01	6.77	8.91	15.02
R7	3.12	10.76	9.46	32.6
R8	4.91	6.78	**35.02**	45.17
R9	4.46	6.94	34.48	**53.67**

Table 4: Results for individual rules for book snippets

Rule	Overall		Conditional	
	EM	DS	EM	DS
R1	6.32	19.19	8.69	26.39
R2	11.26	16.18	30.32	43.56
R3	**12.45**	20.28	34.24	**55.77**
R4	6.91	13.51	18.42	36.0
R5	9.28	**23.87**	15.36	39.47
R6	10.08	16.91	19.31	32.42
R7	9.88	15.21	32.25	49.65
R8	11.26	11.26	**50**	50
R9	11.46	13.28	43.59	50.51

Table 5: Results for individual rules for tweets

This section presents our results in two steps: performance of individual rules that are a part of the rule-based extractor, and performance of the overall approach.

5.3.1. Performance of rules in the rule-based extractor

Tables 4 and 5 present the performance of the rules in our rule-based extractor, for snippets and tweets respectively. The two metrics (exact match accuracy and dice score) are reported for two cases: Overall and Conditional. 'Overall' spans all text units in the dataset whereas 'Conditional' is limited to text units which match a given rule (*i.e.*, where the given linguistic phenomenon of, say, gerunds, etc. is observed). Considering the 'Conditional' case is crucial because a rule may be applicable for a specific form of sarcasm target, but may work accurately in those cases. Such a rule will have a low 'overall exact match/dice score' but a high 'conditional exact match/dice score.' Values in bold indicate the best performing rule for a given performance metric. As seen in the tables, the values for 'conditional' are higher than those for 'Overall'. For example, consider rule R7 in Table 4. Exact match of 3.12 (for overall accuracy) as against 9.46 (for conditional accuracy). This situation is typical of rule-based systems where rules may not cover all cases but be accurate for situations that they do cover. For tweets, R3 has a very high dice score (conditional) (55.77). This rule is based on the intuition in (**?**)

that contrast of positive sentiment with negative situation is a strong indicator of sarcasm target.

5.3.2. Overall performance

Tables 6 and 7 compare the six approaches for snippets and tweets respectively. All our approaches outperform the baseline in the case of exact match and dice score. in the case of tweets, Table 7 shows that the rule-based extractor achieves a dice score of 29.13 while that for statistical extractor is 31.8. Combining the two together (owing to our hybrid architecture) improves the dice score to 39.63. This improvement also holds for book snippets. This **justifies the 'hybrid' nature of our approach**. Hybrid OR performs the best in terms of Dice Score. However, for exact match accuracy, Hybrid AND achieves the best performance (16.51 for snippets and 13.45 for tweets). This is likely because Hybrid AND is restrictive with respect to the predictions it makes for individual words. The statistical extractor performs better than rule-based extractor for all three metrics. For example, in the case of tweets, the dice score for statistical extractor is 31.8 while that for rule-based extractor is 29.13. Also, nearly all results (across approaches and metrics) are higher in the case of tweets as compared to snippets. Since tweets are shorter than snippets (as shown in Table 2), **it is likely that tweets are *more direct* in their ridicule as compared to snippets.** We discuss an experiment to validate this observation in Section 5.4.. Thus, tackling the task of sarcasm target identification and the new dataset we present can help gain insights into the nature of sarcasm.

5.4. Study on nature of sarcasm in tweets versus book snippets

As an example of the kind of investigations into the nature of sarcasm that this dataset can facilitate, we conduct an experiment to test the hypothesis that 'tweets are more direct in their ridicule than snippets'. Book snippets and tweets containing the word 'man' are selected. Thirteen tweets contain the word 'man'. Many book snippets contain the word 'man', of which 13 are randomly chosen among these. 13 tweets and 13 snippets are randomly paired up. Two human annotators (not the ones involved in the sarcasm target annotations) are asked to choose which of the two was more 'direct' in its ridicule. The two human annotators are 25-30 years old, one male engineer (A1) and one female linguist (A2). The two annotators have no prior experience in sarcasm annotation but have studied English as a primary language from school onwards. The two annotators are not told about the claim to be verified, and are not told that one of the pair is a tweet and one of them a book snippet. For every pair, the annotator answers the question *"Which of the two is more direct in its ridicule?"*. A2 selects tweets to be more direct in 11 cases out of 13 while A1 does so in 10 out of 13. In other words, in 11 out of 13 cases, A2 states that for a given pair, the tweet is more direct in its ridicule than the book snippet.

6. Error analysis

We now discuss sources of errors made by our system.

Approach	EM	DC
Baseline 1: All Objective Words	0.0	16.14
Baseline 2: Seq. Labeling	12.05	31.44
Only Rule-Based	9.82	26.02
Only Learning-Based	12.05	31.2
Hybrid OR	7.01	**32.68**
Hybrid AND	**16.51**	21.28

Table 6: Performance of sarcasm target identification for snippets

Approach	EM	DC
Baseline 1: All Objective Words	1.38	27.16
Baseline 2: Seq. Labeling	12.26	33.41
Only Rule-Based	9.48	29.13
Only Learning-Based	10.48	31.8
Hybrid OR	9.09	**39.63**
Hybrid AND	**13.45**	20.82

Table 7: Performance of sarcasm target identification for tweets

- **Confusion between *reason* and target**: Sometimes an action appears to be the target of sarcasm, but sometimes the action (or something else), on deeper reflection, may appear to be only the 'reason' for the speaker to be mocking the actual target via sarcasm. Consider the following examples :

  - For the sentence 'I love *being ignored*', we can only infer that the act of "being ignored" is the target, but in 'I love *being ignored by my girlfriend*', it is not exactly clear if "being ignored" is a part of the target, or here, it just becomes the reason for using sarcasm against the only target : "my girlfriend". It also makes it a bit hard for our system to learn common phrases and utilize patterns for detection.

  - 'I love when I see *people using the elevator at the gym*.' The speaker could be mocking the act of people using the elevator at the gym (by saying that he/she loves to see this act), or could me mocking the people, *because* they use the elevator at a gym!

- **Lack of Context**: Context may often be necessary to determine a sarcasm target. Consider the sentence : '*Oh, you are such a lovely couple! You two could even give those Potters living down the street a real run for their money!*' In this case, you/You two are being ridiculed. For '*those Potters*', two scenarios are possible : a) The Potters could be a well known '*lovely couple*', and their reference is used for augmenting the (sarcastic) praise or b) The Potters could be a couple with a famous *negative reputation*, and their reference was intended to make the listening couple aware of the speaker's sarcastic intentions. In case of the latter scenario, '*those Potters*' becomes a target as well.

Another source of error is cases where **the target lies outside the text**. We now describe such examples and compare the impact of these errors with the overall performance.

In our dataset of book snippets, there are 11 texts (5%) with sarcasm target outside the text. In case of tweets, such cases are much higher: 53 tweets (10%). Table 8 compares the results of our hybrid (OR) approach for the specific case of target being 'outside' the text (indicated by 'Outside cases' in the table), with the results on the complete dataset (indicated by 'Overall' in the table). Dice Score (DS) for book snippets is 6.81 for 'outside' cases as compared to 32.68 for the complete dataset. In general, the performance for the 'outside' cases is lower than the overall performance. This proves the difficulty that the 'Outside' cases presents. The EM and DS values for 'Outside' cases are the same by definition. This is because when the target is 'Outside', a partial match and an exact match are the same. Our approach correctly predicts the label 'Outside' for sentences like '*Yeah, just ignore me. That is TOTALLY the right way to handle this!*' However, our approach gives the incorrect output for some examples. For example, for '*Oh, and I suppose the apples ate the cheese*', the predicted target is not 'Outside' (the expected label) but '*I*'. Similarly, for '*Please keep ignoring me for all of senior year. It's not like we're friends with the exact same people*', the incorrectly predicted target is '*me*' instead of the expected label 'Outside'.

	Book Snippets		Tweets	
	EM	DS	EM	DS
Overall	7.01	32.68	9.09	39.63
'Outside' cases	6.81	6.81	4.71	4.71

Table 8: Comparison of performance of our approach in case of examples with target outside the text (indicated by 'Outside' cases), with complete dataset (indicated by 'Overall'); EM: Exact Match, DS: Dice Score

7. Conclusion & Future work

In this paper, we introduced a novel problem: sarcasm target identification. This problem aims to identify the target of ridicule in a sarcastic text. This target may be a subset of words in the text or a fall-back label 'Outside'. The task poses challenges such as multiple sarcasm targets or sarcasm targets that may not even be present as words in the sentence. We created and released the first dataset which labels the target of ridicule in sarcastic tweets and book snippets. To motivate automatic identification of sarcasm target, we present an introductory approach which is a hybrid of two kinds of extractors: a rule-based and a statistical extractor. An integrator then combines the outputs of the two extractors in two configurations: OR and AND. We evaluate our approach on annotated tweets as well as book snippets. In general, our hybrid OR system performs the best with a Dice score of 39.63. This is higher than two baselines: a naïve baseline designed for the task, and a baseline based on sentiment target identification. Our hybrid approach is also higher than the two extractors individually. This shows that the two extractors collectively form

a good sarcasm target identification approach. We finally present an analysis of errors due to target being outside the text. The dataset and the results obtained from our introductory approach will aid sentiment analysis systems to attribute the negative sentiment expressed via sarcasm to the appropriate entity or aspect.

Our work forms a foundation for future approaches to identify sarcasm targets. As future work, additional rules in the rule-based extractor and novel sets of features in the statistical extractor may be used. Using discourse relations to extract sarcasm targets will be useful. In addition, syntactic dependencies have been found to be useful in case of opinion target extraction (Qiu et al., 2011). Applying these techniques for sarcasm target identification can be useful. A special focus on the 'outside' cases (*i.e.*, cases where the target of ridicule in a sarcastic text is beyond the words present in the sentence) is likely to be helpful for sarcasm target identification.

Bibliography

Altun, Y., Tsochantaridis, I., and Hofmann, T. (2003). Hidden markov support vector machines. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 3–10.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Campbell, J. D. and Katz, A. N. (2012). Are there necessary conditions for inducing a sense of sarcastic irony? *Discourse Processes*, 49(6):459–480.

Davidov, D., Tsur, O., and Rappoport, A. (2010). Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 107–116. Association for Computational Linguistics.

González-Ibánez, R., Muresan, S., and Wacholder, N. (2011). Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2*, pages 581–586. Association for Computational Linguistics.

Jin, W., Ho, H. H., and Srihari, R. K. (2009). Opinionminer: a novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1195–1204. ACM.

Joachims, T. (2006). Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226. ACM.

Joshi, A., Sharma, V., and Bhattacharyya, P. (2015). Harnessing context incongruity for sarcasm detection. In *ACL (2)*, pages 757–762.

Joshi, A., Bhattacharyya, P., and Carman, M. J. (2016a). Automatic sarcasm detection: A survey. *arXiv preprint arXiv:1602.03426*.

Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P., and Carman, M. (2016b). Are word embedding-based fea-

tures useful for sarcasm detection? *arXiv preprint arXiv:1610.00883*.

Khattri, A., Joshi, A., Bhattacharyya, P., and Carman, M. (2015). Your sentiment precedes you: Using an author's historical tweets to predict sarcasm. In *Proceedings of the 6th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 25–30.

Lu, B., Ott, M., Cardie, C., and Tsou, B. K. (2011). Multi-aspect sentiment analysis with topic models. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pages 81–88. IEEE.

McAuley, J. J. and Leskovec, J. (2013). From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*, pages 897–908. ACM.

Michelson, M. and Knoblock, C. A. (2007). Unsupervised information extraction from unstructured, ungrammatical data sources on the world wide web. *International Journal on Document Analysis and Recognition*, 10(3):211–226.

Mukherjee, A. and Liu, B. (2012). Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 339–348. Association for Computational Linguistics.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 115–124. Association for Computational Linguistics.

Pang, B., Lee, L., et al. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Qiu, G., Liu, B., Bu, J., and Chen, C. (2011). Opinion word expansion and target extraction through double propagation. *Computational linguistics*, 37(1):9–27.

Rajadesingan, A., Zafarani, R., and Liu, H. (2015). Sarcasm detection on twitter: A behavioral modeling approach. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 97–106. ACM.

Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., and Huang, R. (2013). Sarcasm as contrast between a positive sentiment and negative situation. In *EMNLP*, volume 13, pages 704–714.

Shamay-Tsoory, S. G., Tomer, R., and Aharon-Peretz, J. (2005). The neuroanatomical basis of understanding sarcasm and its relationship to social cognition. *Neuropsychology*, 19(3):288.

Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Biol. Skr.*, 5:1–34.

Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., and Li, P. (2011). User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1397–1405. ACM.

Titov, I. and McDonald, R. (2008). A joint model of text and aspect ratings for sentiment summarization. *proceedings of ACL-08: HLT*, pages 308–316.

Tsur, O., Davidov, D., and Rappoport, A. (2010). Icwsm-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *ICWSM*, pages 162–169.

Veale, T. and Hao, Y. (2010). Detecting ironic intent in creative comparisons. In *ECAI*, volume 215, pages 765–770.