Harnessing Cognitive Features for Sarcasm Detection

Conference Paper · August 2016

DOI: 10.18653/v1/P16-1104

CITATIONS
39

READS
122

5 authors, including:

Abhijit Mishra
Xavier Institute of Management, Bhubaneswar (XIMB)
74 PUBLICATIONS **2,031** CITATIONS

SEE PROFILE

Diptesh Kanojia
IITB-Monash Research Academy
34 PUBLICATIONS **185** CITATIONS

SEE PROFILE

Seema Nagar
IBM
38 PUBLICATIONS **549** CITATIONS

SEE PROFILE

Pushpak Bhattacharyya
Indian Institute of Technology Bombay
273 PUBLICATIONS **3,379** CITATIONS

SEE PROFILE

Harnessing Cognitive Features for Sarcasm Detection

Abhijit Mishra[†], **Diptesh Kanojia**[†,♣], **Seema Nagar** [⋆], **Kuntal Dey**[⋆],
Pushpak Bhattacharyya[†]
[†]Indian Institute of Technology Bombay, India
[♣]IITB-Monash Research Academy, India
[⋆]IBM Research, India
[†]{abhijitmishra, diptesh, pb}@cse.iitb.ac.in
[⋆]{senagar3, kuntadey}@in.ibm.com

Abstract

In this paper, we propose a novel mechanism for enriching the feature vector, for the task of sarcasm detection, with cognitive features extracted from eye-movement patterns of human readers. Sarcasm detection has been a challenging research problem, and its importance for NLP applications such as review summarization, dialog systems and sentiment analysis is well recognized. Sarcasm can often be traced to *incongruity* that becomes apparent as the full sentence unfolds. This presence of incongruity- implicit or explicit- affects the way readers eyes move through the text. We observe the difference in the behaviour of the eye, while reading sarcastic and non sarcastic sentences. Motivated by this observation, we augment traditional linguistic and stylistic features for sarcasm detection with the cognitive features obtained from readers eye movement data. We perform statistical classification using the enhanced feature set so obtained. The augmented cognitive features improve sarcasm detection by 3.7% (in terms of F-score), over the performance of the best reported system.

1 Introduction

Sarcasm is an intensive, indirect and complex construct that is often intended to express contempt or ridicule [1]. Sarcasm, in speech, is multi-modal, involving tone, body-language and gestures along with linguistic artifacts used in speech. Sarcasm in text, on the other hand, is more restrictive when it comes to such non-linguistic modalities. This makes recognizing textual sarcasm more challenging for both humans and machines.

[1]The Free Dictionary

Sarcasm detection plays an indispensable role in applications like online review summarizers, dialog systems, recommendation systems and sentiment analyzers. This makes automatic detection of sarcasm an important problem. However, it has been quite difficult to solve such a problem with traditional NLP tools and techniques. This is apparent from the results reported by the survey from Joshi et al. (2016). The following discussion brings more insights into this.

Consider a scenario where an online reviewer gives a negative opinion about a movie through sarcasm: *"This is the kind of movie you see because the theater has air conditioning"*. It is difficult for an automatic sentiment analyzer to assign a rating to the movie and, in the absence of any other information, such a system may not be able to comprehend that *prioritizing the air-conditioning facilities of the theater over the movie experience indicates a negative sentiment towards the movie*. This gives an intuition to why, for sarcasm detection, it is necessary to go beyond textual analysis.

We aim to address this problem by exploiting the psycholinguistic side of sarcasm detection, using cognitive features extracted with the help of *eye-tracking*. A motivation to consider cognitive features comes from analyzing human eye-movement trajectories that supports the conjecture: *Reading sarcastic texts induces distinctive eye movement patterns, compared to literal texts.* The cognitive features, derived from human eye movement patterns observed during reading, include two primary feature types:

1. Eye movement characteristic features of readers while reading given text, comprising *gaze-fixaions* (*i.e,*longer stay of gaze on a visual object), forward and backward *saccades* (*i.e.,* quick jumping of gaze between two positions of rest).

2. Features constructed using the statistical and deeper structural information contained in *graph*, created by treating words as vertices and saccades between a pair of words as edges.

The cognitive features, along with textual features used in best available sarcasm detectors, are used to train binary classifiers against given sarcasm labels. Our experiments show significant improvement in classification accuracy over the state of the art, by performing such augmentation.

Feasibility of Our Approach

Since our method requires gaze data from human readers to be available, the methods practicability becomes questionable. We present our views on this below.

Availability of Mobile Eye-trackers

Availability of inexpensive embedded eye-trackers on hand-held devices has come close to reality now. This opens avenues to get eye-tracking data from inexpensive mobile devices from a huge population of online readers non-intrusively, and derive cognitive features to be used in predictive frameworks like ours. For instance, *Cogisen: (http://www.sencogi.com)* has a patent (ID: EP2833308-A1) on "eye-tracking using inexpensive mobile web-cams".

Applicability Scenario

We believe, mobile eye-tracking modules could be a part of mobile applications built for e-commerce, online learning, gaming *etc.* where automatic analysis of online reviews calls for better solutions to detect linguistic nuances like sarcasm. To give an example, let's say a book gets different reviews on Amazon. Our system could watch how readers read the review using mobile eye-trackers, and thereby, decide whether the text contains sarcasm or not. Such an application can horizontally scale across the web and will help in improving automatic classification of online reviews.

Since our approach seeks human mediation, one might be tempted to question the approach of relying upon eye-tracking, an indirect indicator, instead of directly obtaining man-made annotations. We believe, asking a large number of internet audience to annotate/give feedback on each and every sentence that they read online, following a set of annotation instructions, will be extremely intrusive and may not be responded well. Our system, on the other hand, can be seamlessly integrated into existing applications and as the eye-tracking process runs in the background, users will not be interrupted in the middle of the reading. This, thus, offers a more natural setting where human mediation can be availed without intervention.

Getting Users' Consent for Eye-tracking

Eye-tracking technology has already been utilized by leading mobile technology developers (like Samsung) to facilitate richer user experiences through services like *Smart-scroll* (where a user's eye movement determines whether a page has to be scrolled or not) and *Smart-lock* (where user's gaze position decides whether to lock the screen or not). The growing interest of users in using such services takes us to a promising situation where getting users' consent to record eye-movement patterns will not be difficult, though it is yet not the current state of affairs.

Disclaimer: In this work, we focus on detecting sarcasm in *non-contextual* and *short-text* settings prevalent in product reviews and social media. Moreover, our method requires eye-tracking data to be available in the test scenario.

2 Related Work

Sarcasm, in general, has been the focus of research for quite some time. In one of the pioneering works Jorgensen et al. (1984) explained how sarcasm arises when a figurative meaning is used opposite to the literal meaning of the utterance. In the word of Clark and Gerrig (1984), sarcasm processing involves canceling the indirectly negated message and replacing it with the implicated one. Giora (1995), on the other hand, define sarcasm as a mode of indirect negation that requires processing of both negated and implicated messages. Ivanko and Pexman (2003) define sarcasm as a six tuple entity consisting of *a speaker, a listener, Context, Utterance, Literal Proposition* and *Intended Proposition* and study the cognitive aspects of sarcasm processing.

Computational linguists have previously addressed this problem using rule based and statistical techniques, that make use of : **(a)** Unigrams and Pragmatic features (Carvalho et al., 2009; González-Ibánez et al., 2011; Barbieri et al., 2014; Joshi et al., 2015) **(b)** Stylistic patterns (Davidov et al., 2010) and patterns related to *situational disparity* (Riloff et al., 2013) and **(c)** Hastag

interpretations (Liebrecht et al., 2013; Maynard and Greenwood, 2014).

Most of the previously done work on sarcasm detection uses *distant supervision* based techniques (ex: leveraging *hashtags*) and stylistic/pragmatic features (emoticons, laughter expressions such as *"lol" etc*). But, detecting sarcasm in linguistically well-formed structures, in absence of explicit cues or information (like emoticons), proves to be hard using such linguistic/stylistic features alone.

With the advent of sophisticated eye-trackers and electro/magneto-encephalographic (EEG/MEG) devices, it has been possible to delve deep into the cognitive underpinnings of sarcasm understanding. Filik (2014), using a series of eye-tracking and EEG experiments try to show that for unfamiliar ironies, the literal interpretation would be computed first. They also show that a mismatch with context would lead to a re-interpretation of the statement, as being ironic. Camblin et al. (2007) show that in multi-sentence passages, discourse congruence has robust effects on eye movements. This also implies that disrupted processing occurs for discourse incongruent words, even though they are perfectly congruous at the sentence level. In our previous work (Mishra et al., 2016), we augment cognitive features, derived from eye-movement patterns of readers, with textual features to detect whether a human reader has realized the presence of sarcasm in text or not.

The recent advancements in the literature discussed above, motivate us to explore gaze-based cognition for sarcasm detection. As far as we know, our work is the first of its kind.

3 Eye-tracking Database for Sarcasm Analysis

Sarcasm often emanates from *incongruity* (Campbell and Katz, 2012), which enforces the brain to reanalyze it (Kutas and Hillyard, 1980). This, in turn, affects the way eyes move through the text. Hence, **distinctive eye-movement patterns may be observed in the case of successful processing of sarcasm in text in contrast to literal texts**. This hypothesis forms the crux of our method for sarcasm detection and we validate this using our previously released freely available sarcasm dataset[2] (Mishra et al., 2016) enriched with gaze information.

[2]http://www.cfilt.iitb.ac.in/cognitive-nlp

	μ_S	σ_S	μ_NS	σ_NS	t	p
P1	319	145	196	97	14.1	5.84E-39
P2	415	192	253	130	14.0	1.71E-38
P3	322	173	214	160	9.5	3.74E-20
P4	328	170	191	96	13.9	1.89E-37
P5	291	151	183	76	11.9	2.75E-28
P6	230	118	136	84	13.2	6.79E-35
P7	488	268	252	141	15.3	3.96E-43

Table 1: T-test statistics for average fixation duration time per word (in ms) for presence of sarcasm (represented by *S*) and its absence (*NS*) for participants P1-P7.

3.1 Document Description

The database consists of 1,000 short texts, each having 10-40 words. Out of these, 350 are sarcastic and are collected as follows: (a) 103 sentences are from two popular sarcastic quote websites[3], (b) 76 sarcastic short movie reviews are manually extracted from the *Amazon Movie Corpus* (Pang and Lee, 2004) by two linguists. (c) 171 tweets are downloaded using the hashtag *#sarcasm* from Twitter. The 650 non-sarcastic texts are either downloaded from Twitter or extracted from the Amazon Movie Review corpus. The sentences do not contain words/phrases that are *highly* topic or culture specific. The tweets were normalized to make them linguistically well formed to avoid difficulty in interpreting social media lingo. Every sentence in our dataset carries positive or negative opinion about specific "aspects". For example, the sentence *"The movie is extremely well cast"* has positive sentiment about the aspect "cast".

The annotators were seven graduate students with science and engineering background, and possess good English proficiency. They were given a set of instructions beforehand and are advised to seek clarifications before they proceed. The instructions mention the nature of the task, annotation input method, and necessity of head movement minimization during the experiment.

3.2 Task Description

The task assigned to annotators was to read sentences one at a time and label them with with binary labels indicating the polarity (*i.e.,* positive/negative). Note that, the participants were not

[3]http://www.sarcasmsociety.com, http://www.themarysue.com/funny-amazon-reviews

instructed to annotate whether a sentence is sarcastic or not., to rule out the *Priming Effect* (*i.e.,* if sarcasm is expected beforehand, processing incongruity becomes relatively easier (Gibbs, 1986)). The setup ensures its "ecological validity" in two ways: *(1)* Readers are not given any clue that they have to treat sarcasm with special attention. This is done by setting the task to polarity annotation (instead of sarcasm detection). *(2)* Sarcastic sentences are mixed with non sarcastic text, which does not give prior knowledge about whether the forthcoming text will be sarcastic or not.

The eye-tracking experiment is conducted by following the standard norms in eye-movement research (Holmqvist et al., 2011). At a time, one sentence is displayed to the reader along with the "aspect" with respect to which the annotation has to be provided. While reading, an SR-Research Eyelink-1000 eye-tracker (monocular remote mode, sampling rate 500Hz) records several eye-movement parameters like fixations (a long stay of gaze) and saccade (quick jumping of gaze between two positions of rest) and pupil size.

The accuracy of polarity annotation varies between 72%-91% for sarcastic texts and 75%-91% for non-sarcastic text, showing the inherent difficulty of sentiment annotation, when sarcasm is present in the text under consideration. Annotation errors may be attributed to: (a) lack of patience/attention while reading, (b) issues related to text comprehension, and (c) confusion/indecisiveness caused due to lack of context.

For our analysis, we do not discard the incorrect annotations present in the database. Since our system eventually aims to involve online readers for sarcasm detection, it will be hard to segregate readers who misinterpret the text. We make a rational assumption that, for a particular text, most of the readers, from a fairly large population, will be able to identify sarcasm. Under this assumption, the eye-movement parameters, averaged across all readers in our setting, may not be significantly distorted by a few readers who would have failed to identify sarcasm. This assumption is applicable for both regular and multi-instance based classifiers explained in section 6.

4 Analysis of Eye-movement Data

We observe distinct behavior during sarcasm reading, by analyzing the "fixation duration on the text" (also referred to as "dwell time" in the literature) and "scanpaths" of the readers.

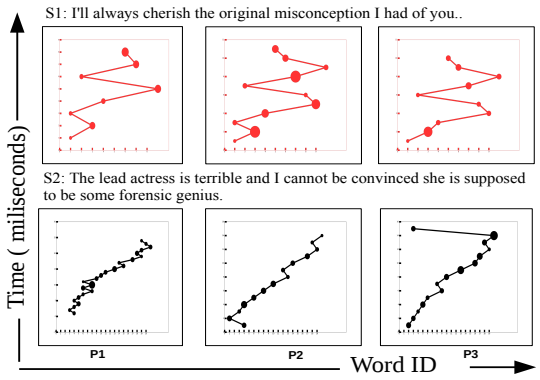

Figure 1: Scanpaths of three participants for two negatively polar sentences sentence *S1* and *S2*. Sentence *S1* is sarcastic but *S2* is not.

4.1 Variation in the Average Fixation Duration per Word

Since sarcasm in text can be expected to induce cognitive load, it is reasonable to believe that it would require more processing time (Ivanko and Pexman, 2003). Hence, fixation duration normalized over total word count should usually be higher for a sarcastic text than for a non-sarcastic one. We observe this for all participants in our dataset, with the *average fixation duration per word* for sarcastic texts being at least 1.5 times more than that of non-sarcastic texts. To test the statistical significance, we conduct a two-tailed t-test (assuming unequal variance) to compare the average fixation duration per word for sarcastic and non-sarcastic texts. The hypothesized mean difference is set to 0 and the error tolerance limit (α) is set to 0.05. The t-test analysis, presented in Table 1, shows that for all participants, a statistically significant difference exists between the average fixation duration per word for sarcasm (higher average fixation duration) and non-sarcasm (lower average fixation duration). This affirms that the presence of sarcasm affects the duration of fixation on words.

It is important to note that longer fixations may also be caused by other linguistic subtleties (such as difficult words, ambiguity and syntactically complex structures) causing delay in comprehension, or occulomotor control problems forcing readers to spend time adjusting eye-muscles. So, an elevated average fixation duration per word may not sufficiently indicate the presence of sarcasm. But we would also like to share that, for our

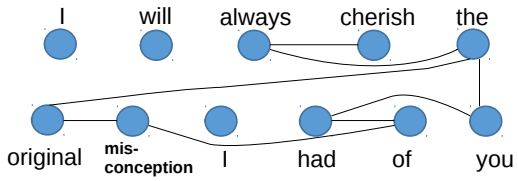

Figure 2: Saliency graph of participant *P1* for the sentence *I will always cherish the original misconception I had of you.*

dataset, when we considered *readability* (Flesch readability ease-score (Flesch, 1948)), *number of words in a sentence* and *average character per word* along with the *sarcasm label* as the predictors of average fixation duration following a linear mixed effect model (Barr et al., 2013), *sarcasm label* turned out to be the most significant predictor with a maximum slope. This indicates that average fixation duration per word has a strong connection with the text being sarcastic, at least in our dataset.

We now analyze *scanpaths* to gain more insights into the sarcasm comprehension process.

4.2 Analysis of Scanpaths

Scanpaths are line-graphs that contain fixations as nodes and saccades as edges; the radii of the nodes represent the fixation duration. A scanpath corresponds to a participant's eye-movement pattern while reading a particular sentence. Figure 1 presents scanpaths of three participants for the sarcastic sentence *S1* and the non-sarcastic sentence *S2*. The x-axis of the graph represents the sequence of words a reader reads, and the y-axis represents a temporal sequence in milliseconds.

Consider a sarcastic text containing incongruous phrases *A* and *B*. Our qualitative scanpath-analysis reveals that scanpaths with respect to sarcasm processing have two typical characteristics. Often, a long *regression* - a saccade that goes to a previously visited segment - is observed when a reader starts reading *B* after skimming through *A*. In a few cases, the fixation duration on *A* and *B* are significantly higher than the average fixation duration per word. In sentence *S1*, we see long and multiple regressions from the two incongruous phrases *"misconception"* and *"cherish"*, and a few instances where phrases *"always cherish"* and *"original misconception"* are fixated longer than usual. Such eye-movement behaviors are not seen for *S2*.

Though sarcasm induces distinctive scanpaths like the ones depicted in Figure 1 in the observed examples, presence of such patterns is not sufficient to guarantee sarcasm; such patterns may also possibly arise from literal texts. We believe that a combination of linguistic features, readability of text and features derived from scanpaths would help discriminative machine learning models learn sarcasm better.

5 Features for Sarcasm Detection

We describe the features used for sarcasm detection in Table 2. The features enlisted under *lexical*,*implicit incongruity* and *explicit incongruity* are borrowed from various literature (predominantly from Joshi et al. (2015)). These features are essential to separate sarcasm from other forms semantic incongruity in text (for example ambiguity arising from *semantic ambiguity* or from *metaphors*). Two additional *textual* features *viz. readability* and *word count* of the text are also taken under consideration. These features are used to reduce the effect of text hardness and text length on the eye-movement patterns.

5.1 Simple Gaze Based Features

Readers' eye-movement behavior, characterized by fixations, forward saccades, skips and regressions, can be directly quantified by simple statistical aggregation (*i.e.,* either computing features for individual participants and then averaging or performing a multi-instance based learning as explained in section 6). Since these eye-movement attributes relate to the cognitive process in reading (Rayner and Sereno, 1994), we consider these as features in our model. Some of these features have been reported by Mishra et al. (2016) for modeling sarcasm understandability of readers. However, as far as we know, these features are being introduced in NLP tasks like textual sarcasm detection for the first time. The values of these features are believed to increase with the increase in the degree of surprisal caused by incongruity in text (except *skip count*, which will decrease).

5.2 Complex Gaze Based Features

For these features, we rely on a graph structure, namely "saliency graphs", derived from eye-gaze information and word sequences in the text.

Constructing Saliency Graphs:

For each reader and each sentence, we construct a "saliency graph", representing the reader's atten-

Subcategory	Feature Name	Type	Intent
Category: Textual Sarcasm Features, Source: Joshi et. al.			
Lexical	Presence of Unigrams (UNI)	Boolean	Unigrams in the training corpus
	Punctuations (PUN)	Real	Count of punctuation marks
Implicit Incongruity	Implicit Incongruity (IMP)	Boolean	Incongruity of extracted implicit phrases (Rilof et.al, 2013)
Explicit Incongruity	Explicit Incongruity (EXP)	Integer	Number of times a word follows a word of opposite polarity
	Largest Pos/Neg Subsequence (LAR)	Integer	Length of the largest series of words with polarities unchanged
	Positive words (+VE)	Integer	Number of positive words
	Negative words (-VE)	Integer	Number of negative words
	Lexical Polarity (LP)	Integer	Sentence polarity found by supervised logistic regression
Category: Cognitive Features. We introduce these features for sarcasm detection.			
Textual	Readability (RED)	Real	Flesch Readability Ease (Flesch, 1948) score of the sentence
	Number of Words (LEN)	Integer	Number of words in the sentence
Simple Gaze Based	Avg. Fixation Duration (FDUR)	Real	Sum of fixation duration divided by word count
	Avg. Fixation Count (FC)	Real	Sum of fixation counts divided by word count
	Avg. Saccade Length (SL)	Real	Sum of saccade lengths (measured by number of words) divided by word count
	Regression Count (REG)	Real	Total number of gaze regressions
	Skip count (SKIP)	Real	Number of words skipped divided by total word count
	Count of regressions from second half to first half of the sentence (RSF)	Real	Number of regressions from second half of the sentence to the first half of the sentence (given the sentence is divided into two equal half of words)
	Largest Regression Position (LREG)	Real	Ratio of the absolute position of the word from which a regression with the largest amplitude (number of pixels) is observed, to the total word count of sentence
Complex Gaze Based	Edge density of the saliency gaze graph (ED)	Real	Ratio of the number of directed edges to vertices in the saliency gaze graph (SGG)
	Fixation Duration at Left/Source (F1H, F1S)	Real	Largest weighted degree (LWD) and second largest weighted degree (SWD) of the SGG considering the fixation duration of word i of edge E_{ij}
	Fixation Duration at Right/Target (F2H, F2S)	Real	LWD and SWD of the SGG considering the fixation duration of word j of edge E_{ij}
	Forward Saccade Word Count of Source (PSH, PSS)	Real	LWD and SWD of the SGG considering the number of forward saccades between words i and j of an edge E_{ij}
	Forward Saccade Word Count of Destination (PSDH, PSDS)	Real	LWD and SWD of the SGG considering the total distance (word count) of forward saccades between words i and j of an edge E_{ij}
	Regressive Saccade Word Count of Source (RSH, RSS)	Real	LWD and SWD of the SGG considering the number of regressive saccades between words i and j of an edge E_{ij}
	Regressive Saccade Word Count of Destination (RSDH, RSDS)	Real	LWD and SWD of the SGG considering the total distance (word count) of regressive saccades between words i and j of an edge E_{ij}

Table 2: The complete set of features used in our system.

tion characteristics. A saliency graph for a sentence S for a reader R, represented as $G = (V, E)$, is a graph with vertices (V) and edges (E) where each vertex $v \in V$ corresponds to a word in S (may not be unique) and there exists an edge $e \in E$ between vertices v_1 and v_2 if R performs at least one saccade between the words corresponding to $v1$ and $v2$.

Figure 2 shows an example of a saliency graph.A saliency graph may be weighted, but not necessarily connected, for a given text (as there may be words in the given text with no fixation on them). The "complex" gaze features derived from saliency graphs are also motivated by the theory of incongruity. For instance, *Edge Density* of a saliency graph increases with the number of distinct saccades, which could arise from the complexity caused by presence of sarcasm. Similarly, the highest weighted degree of a graph is expected to be higher, if the node corresponds to a phrase, incongruous to some other phrase in the text.

6 The Sarcasm Classifier

We interpret sarcasm detection as a binary classification problem. The training data constitutes

Features	P(1)	P(-1)	P(avg)	R(1)	R(-1)	R(avg)	F(1)	F(-1)	F(avg)	Kappa
Multi Layered Neural Network										
Unigram	53.1	74.1	66.9	51.7	75.2	66.6	52.4	74.6	66.8	0.27
Sarcasm (Joshi et. al.)	59.2	75.4	69.7	51.7	80.6	70.4	55.2	77.9	69.9	0.33
Gaze	62.4	76.7	71.7	54	82.3	72.3	57.9	79.4	71.8	0.37
Gaze+Sarcasm	63.4	75	70.9	48	84.9	71.9	54.6	79.7	70.9	0.34
Näive Bayes										
Unigram	45.6	82.4	69.4	81.4	47.2	59.3	58.5	60	59.5	0.24
Sarcasm (Joshi et. al.)	46.1	81.6	69.1	79.4	49.5	60.1	58.3	61.6	60.5	0.25
Gaze	57.3	82.7	73.8	72.9	70.5	71.3	64.2	76.1	71.9	0.41
Gaze+Sarcasm	46.7	82.1	69.6	79.7	50.5	60.8	58.9	62.5	61.2	0.26
Original system by Riloff et.al. : Rule Based with implicit incongruity										
Ordered	60	30	49	50	39	46	54	34	47	0.10
Unordered	56	28	46	40	42	41	46	33	42	0.16
Original system by Joshi et.al. : SVM with RBF Kernel										
Sarcasm (Joshi et. al.)	73.1	69.4	70.7	22.6	95.5	69.8	34.5	80.4	64.2	0.21
SVM Linear: with default parameters										
Unigram	56.5	77	69.8	58.6	75.5	69.5	57.5	76.2	69.6	0.34
Sarcasm (Joshi et. al.)	*59.9*	*78.7*	*72.1*	*61.4*	*77.6*	*71.9*	*60.6*	*78.2*	*72*	*0.39*
Gaze	**65.9**	75.9	72.4	49.7	86	73.2	56.7	80.6	72.2	0.38
Gaze+Sarcasm	63.7	79.5	74	61.7	80.9	74.1	62.7	80.2	74	0.43
Multi Instance Logistic Regression: Best Performing Classifier										
Gaze	65.3	77.2	73	53	**84.9**	73.8	58.5	**80.8**	73.1	0.41
Gaze+Sarcasm	62.5	**84**	**76.5**	**72.6**	76.7	**75.3**	**67.2**	80.2	**75.7**	**0.47**

Table 3: Classification results for different feature combinations. P→ Precision, R→ Recall, F→ F˙score, Kappa→ Kappa statistics show *agreement with the gold labels*. Subscripts 1 and -1 correspond to sarcasm and non-sarcasm classes respectively.

Sentence	Gold	Sarcasm	Gaze	Gaze+Sarcasm
1. I would like to live in Manchester, England. The transition between Manchester and death would be unnoticeable.	S	NS	S	S
2. Helped me a lot with my panic attacks. I took 6 mg a day for almost 20 years. Can't stop of course but it makes me feel very comfortable.	NS	S	NS	NS
3. Forgot to bring my headphones to the gym this morning, the music they play in this gym pumps me up so much!	S	S	NS	NS
4. Best show on satellite radio!! No doubt about it. The little doggy company has nothing even close.	NS	S	NS	S

Table 4: Example test-cases with *S* and *NS* representing labels for sarcastic and not-sarcastic respectively.

994 examples created using our eye-movement database for sarcasm detection. To check the effectiveness of our feature set, we observe the performance of multiple classification techniques on our dataset through a *stratified* 10-fold cross validation. We also compare the classification accuracy of our system and the best available systems proposed by Riloff et al. (2013) and Joshi et al. (2015) on our dataset. Using Weka (Hall et al., 2009) and LibSVM (Chang and Lin, 2011) APIs, we implement the following classifiers:

- Näive Bayes classifier
- Support Vector Machines (Cortes and Vapnik, 1995) with default hyper-paramaters
- Multilayer Feed Forward Neural Network
- Multi Instance Logistic Regression (MILR) (Xu and Frank, 2004)

6.1 Results

Table 3 shows the classification results considering various feature combinations for different classifiers and other systems. These are:

- *Unigram* (with principal components of unigram feature vectors),
- *Sarcasm* (the feature-set reported by Joshi et al. (2015) subsuming unigram features and features from other reported systems)
- *Gaze* (the simple and complex cognitive features we introduce, along with readability and word count features), and
- *Gaze+Sarcasm* (the complete set of features).

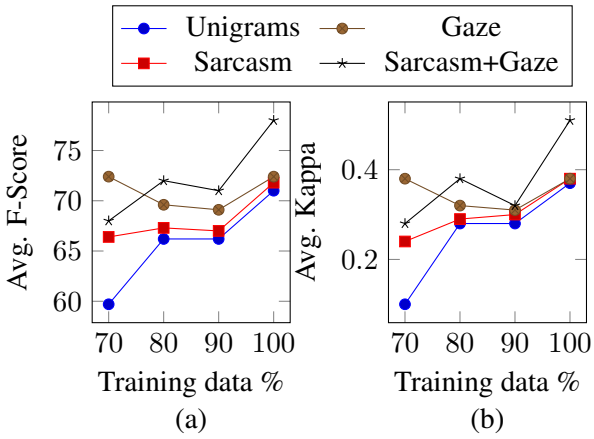

Figure 3: Effect of training data size on classification in terms of (a) F-score and (b) *Kappa* statistics

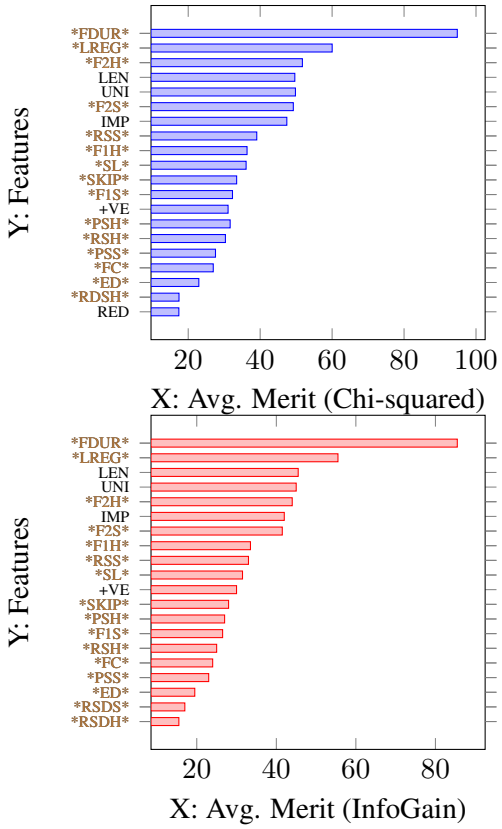

Figure 4: Significance of features observed by ranking the features using *Attribute Evaluation based on Information Gain* and *Attribute Evaluation based on Chi-squared test*. The length of the bar corresponds to the average merit of the feature. Features marked with * are gaze features.

For all regular classifiers, the gaze features are averaged across participants and augmented with linguistic and sarcasm related features. For the MILR classifier, the gaze features derived from each participant are augmented with linguistic features and thus, a multi instance "bag" of features is formed for each sentence in the training data. This multi-instance dataset is given to an MILR classifier, which follows the *standard multi instance assumption* to derive class-labels for each bag.

For all the classifiers, our feature combination outperforms the baselines (considering only unigram features) as well as (Joshi et al., 2015), with the MILR classifier getting an F-score improvement of **3.7%** and *Kappa* difference of **0.08**. We also achieve an improvement of **2%** over the baseline, using SVM classifier, when we employ our feature set. We also observe that the gaze features alone, also capture the differences between sarcasm and non-sarcasm classes with a high-precision but a low recall.

To see if the improvement obtained is statistically significant over the state-of-the art system with textual sarcasm features alone, we perform **McNemar test**. The output of the SVM classifier using only linguistic features used for sarcasm detection by Joshi et al. (2015) and the output of the MILR classifier with the complete set of features are compared, setting threshold $\alpha = 0.05$. There was a significant difference in the classifier's accuracy with **p(two-tailed) = 0.02** with an odds-ratio of **1.43**, showing that the classification accuracy improvement is unlikely to be observed by chance in 95% confidence interval.

6.2 Considering Reading Time as a Cognitive Feature along with Sarcasm Features

One may argue that, considering simple measures of reading effort like "reading time" as cognitive feature instead of the expensive eye-tracking features for sarcasm detection may be a cost-effective solution. To examine this, we repeated our experiments with "reading time" considered as the only cognitive feature, augmented with the textual features. The F-scores of all the classifiers turn out to be close to that of the classifiers considering sarcasm feature alone and the difference in the improvement is not statistically significant ($p > 0.05$). One the other hand, F-scores with gaze features are superior to the F-scores when reading time is considered as a cognitive feature.

6.3 How Effective are the Cognitive Features

We examine the effectiveness of cognitive features on the classification accuracy by varying the input training data size. To examine this, we create a

stratified (keeping the class ratio constant) random train-test split of 80%:20%. We train our classifier with 100%, 90%, 80% and 70% of the training data with our whole feature set, and the feature combination from Joshi et al. (2015). The goodness of our system is demonstrated by improvements in F-score and Kappa statistics, shown in Figure 3.

We further analyze the importance of features by ranking the features based on **(a)** Chi squared test, and **(b)** Information Gain test, using Weka's attribute selection module. Figure 4 shows the top 20 ranked features produced by both the tests. For both the cases, we observe 16 out of top 20 features to be gaze features. Further, in each of the cases, *Average Fixation Duration per Word* and *Largest Regression Position* are seen to be the two most significant features.

6.4 Example Cases

Table 4 shows a few example cases from the experiment with stratified 80%-20% train-test split.

- Example sentence 1 is sarcastic, and requires extra-linguistic knowledge (about poor living conditions at Manchester). Hence, the sarcasm detector relying only on textual features is unable to detect the underlying incongruity. However, our system predicts the label successfully, possibly helped by the gaze features.

- Similarly, for sentence 2, the false sense of presence of incongruity (due to phrases like "Helped me" and "Can't stop") affects the system with only linguistic features. Our system, though, performs well in this case also.

- Sentence 3 presents a false-negative case where it was hard for even humans to get the sarcasm. This is why our gaze features (and subsequently the complete set of features) account for erroneous prediction.

- In sentence 4, gaze features alone false-indicate presence of incongruity, whereas the system predicts correctly when gaze and linguistic features are taken together.

From these examples, it can be inferred that, only gaze features would not have sufficed to rule out the possibility of detecting other forms of incongruity that do not result in sarcasm.

6.5 Error Analysis

Errors committed by our system arise from multiple factors, starting from limitations of the eye-tracker hardware to errors committed by linguistic tools and resources. Also, aggregating various eye-tracking parameters to extract the cognitive features may have caused information loss in the regular classification setting.

7 Conclusion

In the current work, we created a novel framework to detect sarcasm, that derives insights from human cognition, that manifests over eye movement patterns. We hypothesized that distinctive eye-movement patterns, associated with reading sarcastic text, enables improved detection of sarcasm. We augmented traditional linguistic features with cognitive features obtained from readers' eye-movement data in the form of simple gaze-based features and complex features derived from a graph structure. This extended feature-set improved the success rate of the sarcasm detector by 3.7%, over the best available system. Using cognitive features in an NLP Processing system like ours is the first proposal of its kind.

Our general approach may be useful in other NLP sub-areas like sentiment and emotion analysis, text summarization and question answering, where considering textual clues alone does not prove to be sufficient. We propose to augment this work in future by exploring deeper graph and gaze features. We also propose to develop models for the purpose of learning complex gaze feature representation, that accounts for the power of individual eye movement patterns along with the aggregated patterns of eye movements.

Acknowledgments

We thank the members of CFILT Lab, especially Jaya Jha and Meghna Singh, and the students of IIT Bombay for their help and support.

References

Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014. Modelling sarcasm in twitter, a novel approach. *ACL 2014*, page 50.

Dale J Barr, Roger Levy, Christoph Scheepers, and Harry J Tily. 2013. Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of memory and language*, 68(3):255–278.

C. Christine Camblin, Peter C. Gordon, and Tamara Y. Swaab. 2007. The interplay of discourse congruence and lexical association during sentence processing: Evidence from {ERPs} and eye tracking. *Journal of Memory and Language*, 56(1):103 – 128.

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

Paula Carvalho, Luís Sarmento, Mário J Silva, and Eugénio De Oliveira. 2009. Clues for detecting irony in user-generated contents: oh...!! it's so easy;-). In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 53–56. ACM.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Herbert H Clark and Richard J Gerrig. 1984. On the pretense theory of irony.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

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*, pages 107–116. Association for Computational Linguistics.

Hartmut; Wallington Katie; Page Jemma Filik, Ruth; Leuthold. 2014. Testing theories of irony processing using eye-tracking and erps. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 40(3):811–828.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.

Raymond W. Gibbs. 1986. Comprehension and memory for nonliteral utterances: The problem of sarcastic indirect requests. *Acta Psychologica*, 62(1):41 – 57.

Rachel Giora. 1995. On irony and negation. *Discourse processes*, 19(2):239–264.

Roberto González-Ibánez, 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: short papers-Volume 2*, pages 581–586. Association for Computational Linguistics.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. 2011. *Eye tracking: A comprehensive guide to methods and measures*. Oxford University Press.

Stacey L Ivanko and Penny M Pexman. 2003. Context incongruity and irony processing. *Discourse Processes*, 35(3):241–279.

Julia Jorgensen, George A Miller, and Dan Sperber. 1984. Test of the mention theory of irony. *Journal of Experimental Psychology: General*, 113(1):112.

Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015. Harnessing context incongruity for sarcasm detection. *Proceedings of 53rd Annual Meeting of the Association for Computational Linguistics, Beijing, China*, page 757.

Aditya Joshi, Pushpak Bhattacharyya, and Mark James Carman. 2016. Automatic sarcasm detection: A survey. *CoRR*, abs/1602.03426.

Marta Kutas and Steven A Hillyard. 1980. Reading senseless sentences: Brain potentials reflect semantic incongruity. *Science*, 207(4427):203–205.

Christine Liebrecht, Florian Kunneman, and Antal van den Bosch. 2013. The perfect solution for detecting sarcasm in tweets# not. *WASSA 2013*, page 29.

Diana Maynard and Mark A Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Proceedings of LREC*.

Abhijit Mishra, Diptesh Kanojia, and Pushpak Bhattacharyya. 2016. Predicting readers' sarcasm understandability by modeling gaze behavior. In *Proceedings of AAAI*.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics.

Keith Rayner and Sara C Sereno. 1994. Eye movements in reading: Psycholinguistic studies.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *EMNLP*, pages 704–714.

Xin Xu and Eibe Frank. 2004. Logistic regression and boosting for labeled bags of instances. In *Advances in knowledge discovery and data mining*, pages 272–281. Springer.