Harnessing Sequence Labeling for Sarcasm Detection in Dialogue from TV Series ‘Friends’

Aditya Joshi[1,2,3] **Vaibhav Tripathi**[1]
Pushpak Bhattacharyya[1] **Mark Carman**[2]
[1]Indian Institute of Technology Bombay, India
[2]Monash University, Australia
[3]IITB-Monash Research Academy, India
{adityaj, vtripathi,pb}@cse.iitb.ac.in, mark.carman@monash.edu

Abstract

This paper is a novel study that views sarcasm detection in dialogue as a sequence labeling task, where a dialogue is made up of a sequence of utterances. We create a manually-labeled dataset of dialogue from TV series ‘Friends’ annotated with sarcasm. Our goal is to predict sarcasm in each utterance, using sequential nature of a scene. We show performance gain using sequence labeling as compared to classification-based approaches.

Our experiments are based on three sets of features, one is derived from information in our dataset, the other two are from past works. Two sequence labeling algorithms (SVM-HMM and SEARN) outperform three classification algorithms (SVM, Naive Bayes) for all these feature sets, with an increase in F-score of around 4%. Our observations highlight the viability of sequence labeling techniques for sarcasm detection of dialogue.

1 Introduction

Sarcasm is defined as ‘the use of irony to mock or convey contempt’[1]. An example of a sarcastic sentence is ‘*Being stranded in traffic is the best way to start the week*’. In this case, the positive word ‘*best*’ together with the undesirable situation ‘*being stranded in traffic*’ conveys the sarcasm. Because sarcasm has an implied sentiment (negative) that is different from surface sentiment (positive due to presence of ‘best’), it poses a challenge to sentiment analysis systems that aim to determine polarity in text (Pang and Lee, 2008).

Some sarcastic expressions may be more difficult to detect. Consider the possibly sarcastic statement ‘*I absolutely love this restaurant*’. Unlike in the traffic example above, sarcasm in this sentence, if any, can be understood using context which is ‘external’ to the sentence *i.e.,* beyond common world knowledge. [2]. This external context may be available in the conversation that this sentence is a part of. For example, the conversational context may be situational: the speaker discovers a fly in her soup, then looks at her date and says, ‘*I absolutely love this restaurant*’. The conversational context may also be verbal: her date says, ‘*They’ve taken 40 minutes to bring our appetizers*’ to which the speaker responds ‘*I absolutely love this restaurant*’. Both these examples point to the intuition that for dialogue (*i.e.*, data where more than one speaker participates in a discourse), conversational context is often a clue for sarcasm.

For such dialogue, prior work in sarcasm detection (determining whether a text is sarcastic or not) captures context in the form of classifier features such as the topic’s probability of evoking sarcasm, or the author’s tendency to use sarcasm (Rajadesingan et al., 2015; Wallace, 2015). In this paper, we present an alternative hypothesis: **sarcasm detection of dialogue is better formulated as a sequence labeling task, instead of classification task.**

The central message of our work is the efficacy of using sequence labeling as a learning mechanism for sarcasm detection in dialogue, and not in the set of features that we propose for sarcasm detection - although we experiment with three feature sets. For our experiments, we create a manually labeled dataset of dialogues from TV series ‘Friends’. Each dialogue is considered to be a sequence of utterances, and every utterance is annotated as sarcastic or non-sarcastic (Details in Section 3). It may be argued that a TV series episode is dramatized and hence does not reflect real-world conversations. However, although the script of ‘Friends’ is dramatized to suit the situational comedy genre, it takes away nothing from its relevance to real-life conversations except for the volume of sarcastic sentences. Therefore, our findings from this work can, in theory, be reliably extended to work for any real-life utterances. Also, such datasets that are not based on real-world conversations have been used in prior work: emotion detection of children stories in Zhang et al. (2014) and speech transcripts of a MTV show in Rakov and Rosenberg (2013). As a first step in the direction of using sequence labeling, our dataset is a good ‘controlled experiment’ environment (The details are discussed in Section 2). In fact, use of a dataset in a new

[1] As defined by the Oxford Dictionary.

[2] Common world knowledge here refers to a general sentiment map of situations to sentiment. For example, being stranded in traffic is a negative situation to most.

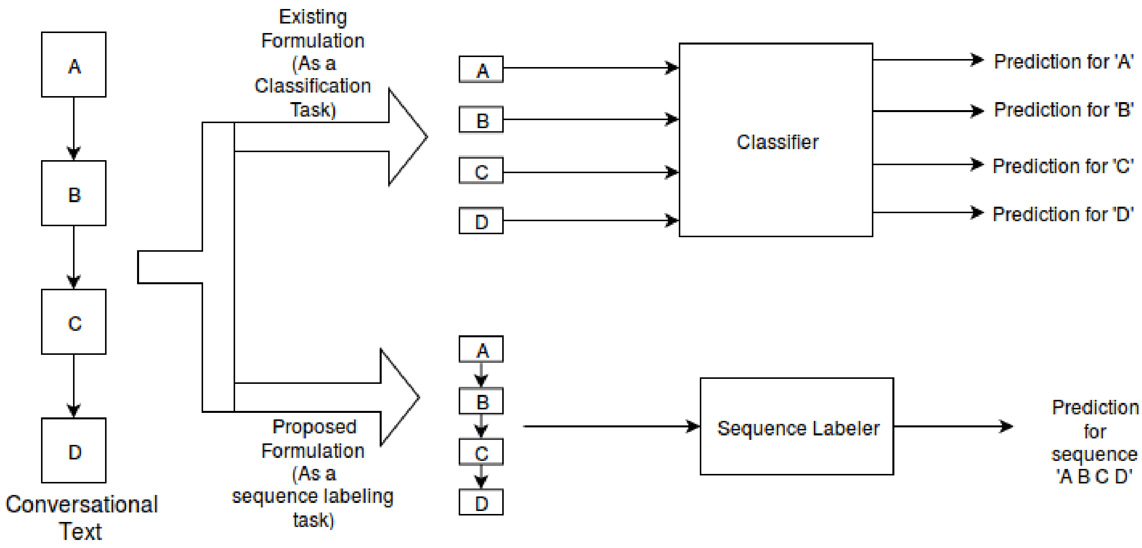

A
B
C
D
Conversational Text
Existing Formulation (As a Classification Task)
Proposed Formulation (As a sequence labeling task)
A
B
C
D
Classifier
Prediction for 'A'
Prediction for 'B'
Prediction for 'C'
Prediction for 'D'
A
B
C
D
Sequence Labeler
Prediction for sequence 'A B C D'

Figure 1: Illustration of our hypothesis for sarcasm detection of conversational text (such as dialogue); A, B, C, D indicate four utterances

genre (TV series transcripts, specifically) has potential for future work in sarcasm detection. Our dataset without the actual dialogues from the show (owing to copyright restrictions) may be available on request.

Based on information available in our dataset (names of speakers, etc.), we present new features. We then compare two sequence labelers (SEARN and SVM-HMM) with three classifiers (SVM with oversampled and undersampled data, and Naïve Bayes), for this set of features and also for features from two prior works. In case of our novel features as well as features reported in prior work, sequence labeling algorithms outperform classification algorithms. There is an improvement of 3-4% in F-score when sequence labelers are used, as compared to classifiers, for sarcasm detection in our dialogue dataset. Since many datasets such as tweet conversations, chat transcripts, etc. are currently available, our findings will be useful to obtain additional contexts in future work.

The rest of the paper is organized as follows. Section 2 motivates the approach and presents our hypothesis. Section 3 describes our dataset, while Section 4 presents the features we use (this includes three configurations: novel features based on our dataset, and features from past work). Experiment setup is in Section 5 and results are given in Section 6. We present a discussion on which types of sarcasm are handled better by sequence labeling and an error analysis in Section 7, and describe related work in Section 8. Finally, we conclude in Section 9.

2 Motivation & Hypothesis

In dialogue, multiple participants take turns to speak. Consider the following snippet from 'Friends' involving two of the lead characters, Ross and Chandler.

[Chandler is at the table. Ross walks in, looking very tanned.]
Chandler: *Hold on! There is something different.*
Ross: *I went to that tanning place your wife suggested.*
Chandler: *Was that place... The Sun?*

Chandler's statement '*Was that place... The Sun?*' is sarcastic. The sarcasm can be understood based on two kinds of contextual information: (a) general knowledge (that sun is indeed hot) (b) Conversational context (In the previous utterance, Ross states that he went to a tanning place). Without information (b), the sarcasm cannot be understood. Thus, dialogue presents a peculiar opportunity: using sequential nature of text for the task at hand.

We hypothesize that '*for sarcasm detection of dialogue, sequence labeling performs better than classification*'. To validate our hypothesis, we consider two feature configurations: (a) novel features designed for our dataset, (b) features as given in two prior works. To further understand where exactly sequence labeling techniques do better, we also present a discussion on which linguistic types of sarcasm benefit the most from sequence labeling in place of classification.

Figure 1 summarizes the scope of this paper. We consider two formulations for sarcasm detection of conversational text. In the first option (i.e. classification), a sequence is broken down into individual instances. One instance as an input to a classification algorithm returns an output for that instance. In the second option (i.e. sequence labeling), a sequence as input to a sequence labeling algorithm returns a sequence of labels as an output. In rest of the paper, we use the following terms:

# Utterances	17338
# Scenes	913
Vocabulary	9345 unigrams
Average Length of Utterance	15.08 words
Average Length of Scene	18.6 utterances

Table 1: Dataset Statistics

1. **Utterance**: An utterance is a contiguous set of sentences spoken by an individual without interruption (from another individual). Every utterance has a speaker, and may be characterized by additional information (such as speaker's expressions and intonation) in the transcript.
2. **Scene/Sequence**: A scene is a sequence of utterances, in which different speakers take turns to speak. We use the terms '*scene*' and '*sequence*' interchangeably.

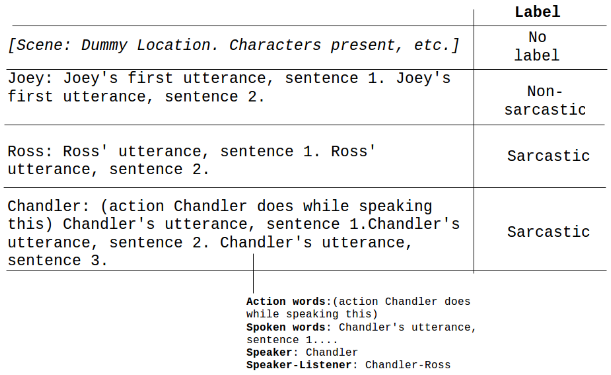
3 '*Friends*' Dataset

Datasets based on literary/creative works have been explored in the past. One such example is emotion classification of children's stories by Zhang Z (2014). Similarly, we create a sarcasm-labeled dataset that consists of transcripts of a comedy TV show, 'Friends[3]' (by Bright/Kauffman/Crane Productions, and Warner Bros. Entertainment Inc.). We download these transcripts from OpenSubtitles[4] as given by Lison and Tiedemann (2016), with additional pre-processing from a fan-contributed website called `http://www.friendstranscripts.tk`. Each scene begins with a description of the location/situation followed by a series of utterances spoken by characters. Figure 2 shows an illustration of our dataset. This is (obviously) a dummy example that has been anonymized.

The reason behind choosing a TV show transcript as our dataset was to restrict to a small set of characters (so as to leverage on speaker-specific features) that use a lot of humor. These characters are often sarcastic towards each other because of their inter-personal relationships. In fact, past linguistic studies also show how sarcasm is more common between familiar speakers, and often friends (Gibbs, 2000). A typical snippet is:

[Scene: Chandler and Monica's room. Chandler is packing when Ross knocks on the door and enters...]
Ross: Hey!
Chandler: Hey!
Ross: You guys ready to go?
Chandler: Not quite. Monica's still at the salon, and I'm just finishing packing.

Our annotators are linguists with an experience of more than 8k hours of annotation, and are not authors of this paper. A complete scene is visible to the annotators at a time, so that they understand complete context of the scene. They perform the task of annotating every utterance in this scene with two labels: sarcastic and non-sarcastic. The two annotators separately perform this annotation over multiple sessions. To minimize bias beyond the scope of this annotation, we selected annotators who had never watched Friends before this annotation task.



	Label
[Scene: Dummy Location. Characters present, etc.]	No label
Joey: Joey's first utterance, sentence 1. Joey's first utterance, sentence 2.	Non-sarcastic
Ross: Ross' utterance, sentence 1. Ross' utterance, sentence 2.	Sarcastic
Chandler: (action Chandler does while speaking this) Chandler's utterance, sentence 1.Chandler's utterance, sentence 2. Chandler's utterance, sentence 3.	Sarcastic

Action words:(action Chandler does while speaking this)
Spoken words: Chandler's utterance, sentence 1....
Speaker: Chandler
Speaker-Listener: Chandler-Ross

Figure 2: Example from our Dataset: Part of a Scene

The annotations [5] may be available on request, subject to copyright restrictions. Every utterance is annotated with a label while description of a scene is not annotated.

The inter-annotator agreement for a subset of 105 scenes[6] (around 1600 utterances) is 0.44. This is comparable with other manually annotated datasets in sarcasm detection (Tsur et al., 2010). Table 1 shows the relevant statistics of the complete dataset (in addition to 105 scenes as mentioned above). There are 17338 utterances in 913 scenes. Out of these, 1888 utterances are labeled as sarcastic. Average length of a scene is 18.6 utterances.

Table 2 shows additional statistics. Table 2(a) shows that Chandler is the character with highest proportion of sarcastic utterances (22.24%). Table 2(b) shows that sarcastic utterances have higher surface positive word score[7] (1.55) than non-sarcastic (0.97) or overall utterances (1.03). This validates the past observation that sarcasm is often expressed through positive words (and sometimes contrasted with negation)(Joshi et al., 2015). Finally, Table 2(c) shows that sarcastic utterances also have higher proportion of non-verbal indicators (action words) (28.23%) than non-sarcastic or overall utterances.

[3] http://www.imdb.com/title/tt0108778/

[4] `http://www.opensubtitles.org`

[5] without textual content, keeping in view copyright restrictions.

[6] For these scenes, the annotators later discussed and arrived at a consensus- they were then added to the dataset. The remaining scenes are done by either of the two annotators.

[7] This is computed using a simple lexicon lookup, as in case of conversational context features below.

Character	% sarcastic
Phoebe	9.70
Joey	11.05
Rachel	9.74
Monica	8.87
Chandler	22.24
Ross	8.42

	Surface Positive Sentiment Score	Surface Negative Sentiment Score
Sarcastic	1.55	1.20
Non-sarcastic	0.97	0.75
All	1.03	0.79

	Actions (%)
Sarcastic	28.23
Non-sarcastic	23.95
All	24.43

Table 2: Dataset statistics related to: (a) percentage of sarcastic utterances for six lead characters, (b) average surface positive and negative scores for the two classes, (c) percentage of sarcastic and non-sarcastic utterances with actions

Feature	Description
Lexical Features	
Spoken words	Unigrams of spoken words
Conversational Context Features	
Actions	Unigrams of action words
Sentiment Score	Positive & Negative score of utterance
Previous Sentiment Score	Positive & Negative score of previous utterance in the sequence
Speaker Context Features	
Speaker	Speaker of this utterance
Speaker-Listener	Pair of speaker of this utterance and speaker of the previous utterance

Table 3: Our Dataset-Derived Features

4 Features

To ensure that our hypothesis is not dependent on choice of features, we show our results on two configurations: (a) when dataset-derived features (*i.e.*, novel features designed based on our dataset) are used, and (b) when features reported in prior work are used. We describe these in forthcoming subsections.

4.1 Dataset-derived Features

We design our dataset-derived features based on information available in our dataset. An utterance consists of three parts:

1. **Speaker**: The name of the speaker is the first word of an utterance, and is followed by a colon. In case of the second utterance in Figure 2, the speaker is '*Ross*' while in the third, the speaker is '*Chandler*'.

2. **Spoken words**: This is the textual portion of what the speaker says. In the second utterance in Figure 2, the spoken words are '*Chandler's utterance, sentence 1..*'.

3. **Action words**: Actions that a speaker performs while speaking the utterance are indicated in parentheses. These are useful clues that form additional context. Unlike speaker and spoken words, action words may or may not be present. In the second utterance in Figure 2, there are no action words while in the third utterance, '*action Chandler does while reading this*' are action words.

Based on this information, we design three categories of features (listed in Table 3). These are:

1. **Lexical Features**: These are unigrams in the spoken words. We experimented with both count and boolean representations, and the results are comparable. We report values for boolean representation.

2. **Conversational Context Features**: In order to capture conversational context, we use three kinds of features. *Action words* are unigrams indicated within parentheses. The intuition is that a character 'raising her eyebrows' (action) is different from saying "raising her eyebrows". As the next feature, we use *sentiment score* of this utterance. These are two values: positive and negative scores. These scores are the positive and negative words present in an utterance. The third kind of conversational context features is the *sentiment score of the previous utterance*. This captures phenomena such as a negative remark from one character eliciting sarcasm from another. This is similar to the situation described in Joshi et al. (2015). Thus, for the third utterance in Figure 2, the sentiment score of Chandler's utterance forms the *Sentiment score* feature, while that of Ross's utterance forms *Sentiment score of previous utterance*.

3. **Speaker Context Features**: We use name of the speaker and name of the speaker-listener pair as features. The listener is assumed to be the speaker of the previous utterance in the sequence[8]. The speaker feature aims to capture the sarcastic nature of each of these characters, while

[8] The first utterance in a sequence has a null value for previous speaker.

the speaker-listener feature aims to capture interpersonal interactions between different characters. In the context of third utterance in Figure 2, the speaker is '*Chandler*' while speaker-listener pair is '*Chandler-Ross*'.

4.2 Features from Prior Work

We also compare our results with features presented in two prior works[9]:

1. **Features given in González-Ibánez et al. (2011)**: These features are: (a) Interjections, (b) Punctuations, (c) Pragmatic features (where we include action words as well), (d) Sentiment lexicon-based features from LIWC (Pennebaker et al., 2001) (where they include counts of linguistic process words, positive/negative emotion words, etc.).
2. **Features given in Buschmeier et al. (2014)**: In addition to unigrams, the features used by them are: (a) Hyperbole (captured by three positive or negative words in a row), (b) Quotation marks and ellipsis, (c) Positive/Negative Sentiment Scores followed by punctuation (this includes more than one positive or negative words with an exclamation mark or question mark at the end), (d) Positive/Negative Sentiment Scores followed by ellipsis (this includes more than one positive or negative words with a '...' at the end, (e) Punctuation, (f) Interjections, and (g) Laughter expressions (such as 'haha').

5 Experiment Setup

We experiment with three classification techniques and two sequence labeling techniques:

1. **Classification Techniques**: We use Naïve Bayes and SVM as classification techniques. Naïve Bayes implementation provided in Scikit (Pedregosa et al., 2011) is used. For SVM, we use SVM-Light (Joachims, 1999). Since SVM does not do well for datasets with a large class imbalance (Akbani et al., 2004)[10], we use sampling to deal with this skew as done in Kotsiantis et al. (2006). We experiment with two configurations:
    - SVM (Oversampled) *i.e.*, **SVM (O)**: Sarcastic utterances are duplicated to match the count of non-sarcastic utterances.
    - SVM (Undersampled) *i.e.*, **SVM (U)**: Random non-sarcastic utterances are dropped to match the count of sarcastic utterances.
2. **Sequence Labeling Techniques**: We use **SVM-HMM** by Altun et al. (2003) and **SEARN** by Daumé III et al. (2009). SVM-HMM is a sequence labeling algorithm that combines Support Vector Machines and Hidden Markov Models. SEARN is a sequence labeling algorithm that integrates search and learning to solve prediction problems. The implementation of SEARN that we use relies on perceptron as the base classifier. Daumé III et al. (2009) show that SEARN outperforms other sequence labeling techniques (such as CRF) for tasks like character recognition and named entity class identification.

Thus, we wish to validate our hypothesis in case of:

1. Our data-derived features as given in Section 4.1.
2. Past features from González-Ibánez et al. (2011) and Buschmeier et al. (2014) as given in Section 4.2.

Algorithm	**Precision (%)**	**Recall (%)**	**F-Score (%)**
Formulation as Classification			
SVM (U)	83.6	48.6	57.2
SVM (O)	**84.4**	76.8	79.8
Naïve Bayes	77.2	33.8	42
Formulation as Sequence Labeling			
SVM-HMM	83.8	**88.2**	**84.2**
SEARN	82.6	83.4	82.8

Table 4: Comparison of sequence labeling techniques with classification techniques, for features reported in dataset-derived features

We report **weighted average values of precision, recall and F-score** computed using five-fold cross-validation for all experiments, and class-wise precision, recall, F-score wherever necessary. The folds are created on the basis of sequences and not utterances. This means that a sequence does not get split across different folds.

6 Results

Section 6.1 describes performance of traditional models that use dataset-derived features (as given in Section 4.1), while Section 6.2 does so for features from prior work (as given in Section 4.2).

6.1 Performance on Dataset-derived Features

Table 4 compares the performance of the two formulations: classification and sequence labeling, for our dataset-derived features. When classification techniques are used, we obtain the best F-score of 79.8% with SVM (O). However, when sequence labeling techniques are used, the best F-score is 84.2%. In terms of

[9] The two prior works are chosen based on what information was available in our dataset for the purpose of re-implementation. For example, approaches that use the Twitter profile information or the follower/friends structure in the Twitter, cannot be computed for our dataset.

[10] We also observe the same.

	Sarcastic			Non-Sarcastic		
	Precision	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
SVM (U)	14	**68.8**	23	**92.2**	46.2	61.6
SVM (O)	22.4	44	**29**	91.8	81	86
Naive Bayes	9.8	59.8	16.8	85.8	30.6	45
SVM-HMM	**35.8**	7.6	12.6	89.4	**98.2**	**93.6**
SEARN	22.2	19.4	20	90	91.6	90.4

Table 5: Class-wise precision/recall values for all techniques using our dataset-derived features

F-score, **our two sequence labeling techniques perform better than all three classification techniques**. The increase in recall is high - the best value for classification techniques is (SVM (O)) 76.8%, while that for sequence labeling techniques (SVM-HMM) is 88.2%. It must be noted that sentiment of previous utterance is one of the features of both the classification and sequence labeling techniques. Despite that, sequence labeling techniques perform better.

Alg.	Feature (Best)	P (%)	R (%)	F (%)
Formulation as Classification				
SVM (U)	Unigram+ Spkr-Listnr+ Action+ Senti. Score	**84.8**	49.4	57.4
SVM (O)	Unigram+ Speaker+ Spkr-Listnr+Senti. Score	84	79	81.2
Naïve Bayes	All features	77.2	33.8	42
Formulation as Sequence Labeling				
SVM-HMM	Unigram+ Speaker+ Spkr-Listnr+ Prev. Senti. Score + Action	83.2	**87.8**	**84.4**
SEARN	All features	82.6	83.4	82.8

Table 6: Feature Combinations for which different techniques exhibit their best performance for dataset-derived features

Table 5 shows class-wise precision/recall values for these techniques. The best value of precision for sarcastic class is obtained in case of SVM-HMM, *i.e.*, 35.8%. The best F-score for the sarcastic class is in the case of SVM (O) (29%) whereas that for the non-sarcastic class is in the case of SVM-HMM (93.6%). Tables 4 and 5 show that it is due to a high recall, sequence labeling techniques perform better than classification techniques.

It may be argued that the benefit in case of sequence labeling is due to our features, and is not a benefit of the sequence labeling formulation itself. Hence, we ran these five techniques with all possible combinations of features. Table 6 shows the best performance obtained by each of the classifiers, and the corresponding (best) feature combinations. The table can be read as: SVM (O) obtains a F-score of 81.2% when spoken words, speaker, speaker-listener and sentiment score are used as features. The table shows that **even if we consider the best performance of each of the techniques (with different feature sets), classifiers are not able to perform as well as sequence labeling**. The best sequence labeling algorithm (SVM-HMM) gives a F-score of 84.4% while the best classifier (SVM(O)) has an F-score of 81.2%. We emphasize that both SVM-HMM and SEARN have higher recall values than the three classification techniques.

These findings show that **for our novel set of dataset-derived features, sequence labeling works better than classification**.

6.2 Performance on Features Reported in Prior Work

We now show our evaluation on two sets of features reported in prior work. These sets of features as given in two prior works by Buschmeier et al. (2014) and González-Ibánez et al. (2011).

Table 7 compares classification techniques with sequence labeling techniques for features given in González-Ibánez et al. (2011)[11]. Table 8 shows corresponding values for features given in Buschmeier et al. (2014)[12]. For features by González-Ibánez et al. (2011), SVM (O) gives the best F-score for classification techniques (79%), whereas SVM-HMM shows an improvement of 4% over that value. Recall **increases by 11.8%** when sequence labeling techniques are used instead of classification.

In case of features by Buschmeier et al. (2014), the improvement in performance achieved by using sequence labeling as against classification is 2.8%. The best recall for classification techniques is 77.8% (for SVM (O)). In this case as well, the recall increases by 10% for sequence labeling.

These findings show that **for two feature sets reported in prior work, sequence labeling works bet-**

[11] The paper reports best accuracy of 65.44% for their dataset. This shows that our implementation is competent.

[12] The paper reports best F-score of 67.8% for their dataset. This shows that our implementation is competent.

ter than classification.

Algorithm	P (%)	R (%)	F (%)
Features from Gonzalez-Ibanez et al. (2011)			
Formulation as Classification			
SVM (U)	**86.4**	26	27
SVM (O)	84.6	75.6	79
Naive Bayes	77.2	43.8	48.4
Formulation as Sequence Labeling			
SVM-HMM	83.4	**87.4**	**83**
SEARN	82	82.4	81.8

Table 7: Comparison of sequence labeling techniques with classification techniques, for features reported in Gonzalez-Ibanez et al. (2011)

Algorithm	P (%)	R (%)	F (%)
Features from Buschmeier et al. (2014)			
Formulation as Classification			
SVM (U)	**85.4**	40.6	46.8
SVM (O)	84.6	77.8	80.4
Naïve Bayes	76.6	27.2	32.8
Formulation as Sequence Labeling			
SVM-HMM	84.2	**87.8**	**83.2**
SEARN	82.4	83.8	82.4

Table 8: Comparison of sequence labeling techniques with classification techniques, for features reported in Buschmeier et al. (2014)

7 Discussion

In previous sections, we show that quantitatively, sequence labeling techniques perform better than classification techniques. In this section, we delve into the question: '*What does this improved performance mean, in terms of forms of sarcasm that sequence labeling techniques are able to handle better than classification?*' To understand the implication of using sequence labeling, we randomly select 100 examples that were correctly labeled by sequence labeling techniques but incorrectly labeled by classification techniques. Our annotators manually annotated them into one among four categories of sarcasm as given in Camp (2012). Table 9 shows the proportion of these utterances. Like-prefixed and illocutionary sarcasm types are the ones that require context for understanding sarcasm. We observe that around 71% of our examples belong to these two types of sarcasm. This means that **our intuition that sequence labeling will better capture conversational context reflects in the forms of sarcasm for which sequence labeling improves over classification.**

On the other hand, examples where our system makes errors can be grouped as:

- **Topic Drift**: Eisterhold et al. (2006) state that topic change/drift is a peculiarity of sarcasm. For example, when Phoebe gets irritated with another character talking for a long time, she says,"See? Vegetarianism benefits everyone". This was misclassified by our system.
- **Short expressions**: Short expressions occurring in the context of a conversation may express sarcasm. Expressions such as "*Oh God, is it*" and "*Me too*" were misclassified as non-sarcastic. However, in the context of the scene, these were sarcastic utterances.
- **Dry humor**: In the context of a conversation, sarcasm may be expressed in response to a long serious description. Our system was unable to capture such sarcasm in some cases. When a character gives long description of advantages of a particular piece of clothing, Chandler asks sarcastically, "*Are you aware that you're still talking?*".
- **Implications in popular culture**: The utterance "*Ok, I smell smoke. Maybe that's cause someone's pants are on fire*" was misclassified by our system. The popular saying 'Liar, liar, pants on fire[13]' was the context that was missing in our case.
- **Background knowledge**: When a petite girl walks in, Rachel says "*She is so cute! You could fit her right in your little pocket*".
- **Long-range connection**: In comedy shows like Friends, humor is often created by introducing a concept in the initial part and then repeating it as an impactful, sarcastic remark. For example, in beginning of an episode, Ross says that he has never grabbed a spoon before - and at the end of the episode, he says with a sarcastic tone "*I grabbed a spoon*".
- **Incongruity with situation in the scenes**: Utterances that were incongruent with non-verbal situations could not be adequately identified. For example, Ross enters an office wearing a piece of steel bandaged to his nose. In response, the receptionist says, "Oh, that's attractive".
- **Sarcasm as a part of a longer sentence**: In several utterances, sarcasm is a subset of a longer sentence, and hence, the non-sarcastic portion may dominate the rest of the sentence.

These errors point to future directions in which sequence labeling algorithms may be optimized to improve their impact on sarcasm detection.

[13] http://www.urbandictionary.com/define.php?term=Liar %20Liar%20Pants%20On%20Fire

Type	Examples (%)
Propositional	14.28
Embedded	4.08
Illocutionary	**40.81**
Like-prefixed	**31.63**
Other	9.18

Table 9: Proportion of utterances of different types of sarcasm that were correctly labeled by sequence labeling but incorrectly labeled by classification techniques

8 Related Work

Sarcasm detection approaches using different features have been reported (Tepperman et al., 2006; Kreuz and Caucci, 2007; Tsur et al., 2010; Davidov et al., 2010; Veale and Hao, 2010; González-Ibánez et al., 2011; Reyes et al., 2012; Joshi et al., 2015; Buschmeier et al., 2014). However, Wallace et al. (2014) show how context beyond the target text (*i.e.*, extra-textual context) is necessary for humans as well as machines, in order to identify sarcasm. Following this, the new trend in sarcasm detection is to explore the use of such extra-textual context (Khattri et al., 2015; Rajadesingan et al., 2015; Bamman and Smith, 2015; Wallace, 2015). (Wallace, 2015) uses meta-data about reddits to predict sarcasm in a reddit[14] comment. (Rajadesingan et al., 2015) present a suite of classifier features that capture different kinds of context: context related to the author, conversation, etc. The new trend in sarcasm detection is, thus, to look at additional context beyond the text where sarcasm is to be predicted.

The work closest to ours is by Wang et al. (2015). They use a labeled dataset of 1500 tweets, the labels for which are obtained **automatically**. Due to their automatically labeled gold dataset and their lack of focus on labeling utterances in a sequence, our analysis seems to be more rigorous. Our work substantially differs from theirs: (a) They do not deal with dialogue, (b) Their goal is to predict sarcasm of a tweet, using series of past tweets as the context *i.e.*, only the last tweet in the sequence. Our goal is to predict sarcasm in *every* element of the sequence: a lot more rigorous task. **Note that the two differ in the way precision/recall values will be computed**. (c) **Their 'gold' standard dataset is annotated by an automatic classifier. On the other hand, every textual unit (utterance) in our gold standard dataset is manually labeled - making our dataset and hence, findings lot more reliable.** (c) They consider three types of sequences: conversational, historical and topic-based. Historical context is series of tweets by this author, while topic-based context is series of tweets containing a hashtag in the tweet to be classified. We do not use the two because they do not seem suitable for our dataset. They show that a sequence labeling algorithm works well to detect sarcasm of a tweet with a pseudo-sequence generated using such additional context. They attempt to obtain correct prediction only for a single target tweet with no consideration to other elements in the context, which is completely different from our goal. They do not bother about other elements in the sequence but only *use* an algorithm to perform sarcasm detection of a tweet.

Several approaches for sequence labeling in sentiment classification have been studied. Zhao et al. (2008) perform sentiment classification using conditional random fields. Zhang et al. (2014) deal with emotion classification. Using a dataset of children's stories manually annotated at the sentence level, they employ HMM to identify sequential structure and a classifier to predict emotion in a particular sentence. Mao and Lebanon (2006) present a isotonic CRF that predicts global and local sentiment of documents, with additional mechanism for author-specific distributions and smoothing sentiment curves. Yessenalina et al. (2010) present a joint learning algorithm for sentence-level subjectivity labeling and document-level sentiment labeling. Choi and Cardie (2010) deal with sequence learning to jointly identify scope of opinion polarity expressions, and polarity labels. Taking inspiration from use of sequence labeling for sarcasm detection, our work takes the first step to show if sequence labeling techniques are helpful at all. They experiment with MPQA corpus that is labeled at the sentence level for polarity as well as intensity. Specialized sequence labeling techniques like these are the next step to our first step: showing if sequence labeling techniques are helpful at all, for sarcasm detection of dialogue.

9 Conclusion & Future Work

We explored how sequence labeling can be used for sarcasm detection of dialogue. We formulated sarcasm detection of dialogue as a task of labeling each utterance in a sequence, with one among two labels: sarcastic and non-sarcastic. For our experiments, we created a manually annotated dataset of transcripts from a popular TV show 'Friends'. Our dataset consisted of 913 scenes where every utterance was annotated as sarcastic or not.

We experiment with: (a) a novel set of features derived from our dataset, (b) sets of features from two prior works. Our dataset-derived features are: (a) lexical features, (b) conversational context features, and (c) author context features. Using these features, we compared two classes of learning techniques: classifiers (SVM (undersampled), SVM (oversampled) and Naïve Bayes) and sequence labeling techniques (SVM-HMM and SEARN). For our classifiers, the best F-score was obtained with SVM (O) (i.e. 79.8%) while the best F-score for sequence labeling techniques was obtained using SVM-HMM (i.e. 84.2%). Even in case of the **best combinations** of our features for each algorithm, both sequence labeling techniques outperformed the classifiers. In addition, we also experimented with

[14] `www.reddit.com`

features introduced in two prior works. We observed an improvement of 2.8% for features in Buschmeier et al. (2014) and 4% for features in González-Ibánez et al. (2011) when sequence labeling techniques were used as against classifiers. In all cases, **sequence labeling techniques had a substantially high recall as compared to classification techniques (10% in case of Buschmeier et al. (2014), 12% in case of González-Ibánez et al. (2011))**. To understand which forms of sarcasm get correctly labeled by sequence labeling (and not by classification), we manually evaluated 100 examples. 71% of these examples consisted of sarcasm that could be understood only with conversational context. Our error analysis points to interesting future work for sarcasm detection of dialogue such as long-range connection, lack of conversational clues, and sarcasm a part of long utterances.

Thus, we observe that for sarcasm detection of our dataset, in case of different feature configurations, sequence labeling performs better than classification. **Our observations establish the efficacy of sequence labeling techniques for sarcasm detection of dialogue.**

Future work on repeating these experiments for other forms of dialogue (such as twitter conversations, chat transcripts, etc.) is imperative. Also, a combination of unified sarcasm and emotion detection using sequence labeling is another promising line of work. It would be interesting to see if deep learning-based models that perform sequence labeling perform better than those that perform classification.

Acknowledgment

We express our gratitude towards our annotators, Rajita Shukla and Jaya Saraswati. We also thank Prerana Singhal for her support. Aditya's PhD is funded by TCS Research Scholar Fellowship.

References

Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. 2004. Applying support vector machines to imbalanced datasets. In *Machine learning: ECML 2004*, pages 39–50. Springer.

Yasemin Altun, Ioannis Tsochantaridis, Thomas Hofmann, et al. 2003. Hidden markov support vector machines. In *ICML*, volume 3, pages 3–10.

David Bamman and Noah A Smith. 2015. Contextualized sarcasm detection on twitter. In *Ninth International AAAI Conference on Web and Social Media*.

Konstantin Buschmeier, Philipp Cimiano, and Roman Klinger. 2014. An impact analysis of features in a classification approach to irony detection in product reviews. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 42–49.

Elisabeth Camp. 2012. Sarcasm, pretense, and the semantics/pragmatics distinction*. *Noûs*, 46(4):587–634.

Yejin Choi and Claire Cardie. 2010. Hierarchical sequential learning for extracting opinions and their attributes. In *Proceedings of the ACL 2010 conference short papers*, pages 269–274. Association for Computational Linguistics.

Hal Daumé III, John Langford, and Daniel Marcu. 2009. Search-based structured prediction.

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.

Jodi Eisterhold, Salvatore Attardo, and Diana Boxer. 2006. Reactions to irony in discourse: Evidence for the least disruption principle. *Journal of Pragmatics*, 38(8):1239–1256.

Raymond W Gibbs. 2000. Irony in talk among friends. *Metaphor and symbol*, 15(1-2):5–27.

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.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, chapter 11, pages 169–184. MIT Press, Cambridge, MA.

Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015. Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, volume 2, pages 757–762.

Anupam Khattri, Aditya Joshi, Pushpak Bhattacharyya, and Mark James Carman. 2015. Your sentiment precedes you: Using an authors historical tweets to predict sarcasm. In *6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, page 25.

Sotiris Kotsiantis, Dimitris Kanellopoulos, Panayiotis Pintelas, et al. 2006. Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, 30(1):25–36.

Roger J Kreuz and Gina M Caucci. 2007. Lexical influences on the perception of sarcasm. In *Proceedings of the Workshop on computational approaches to Figurative Language*, pages 1–4. Association for Computational Linguistics.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *Proceedings of LREC 2016*.

Yi Mao and Guy Lebanon. 2006. Isotonic conditional random fields and local sentiment flow. In *Advances in neural information processing systems*, pages 961–968.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830.

James W Pennebaker, Martha E Francis, and Roger J Booth. 2001. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71:2001.

Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 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.

Rachel Rakov and Andrew Rosenberg. 2013. " sure, i did the right thing": a system for sarcasm detection in speech. In *INTERSPEECH*, pages 842–846.

Antonio Reyes, Paolo Rosso, and Davide Buscaldi. 2012. From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering*, 74:1–12.

Joseph Tepperman, David R Traum, and Shrikanth Narayanan. 2006. " yeah right": sarcasm recognition for spoken dialogue systems. In *INTERSPEECH*. Citeseer.

Oren Tsur, Dmitry Davidov, and Ari Rappoport. 2010. Icwsm-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *ICWSM*.

Tony Veale and Yanfen Hao. 2010. Detecting ironic intent in creative comparisons. In *ECAI*, volume 215, pages 765–770.

Byron C Wallace, Laura Kertz Do Kook Choe, and Eugene Charniak. 2014. Humans require context to infer ironic intent (so computers probably do, too). In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 512–516.

Byron C Wallace. 2015. Sparse, contextually informed models for irony detection: Exploiting user communities,entities and sentiment. In *ACL*.

Zelin Wang, Zhijian Wu, Ruimin Wang, and Yafeng Ren. 2015. Twitter sarcasm detection exploiting a context-based model. In *Web Information Systems Engineering–WISE 2015*, pages 77–91. Springer.

Ainur Yessenalina, Yisong Yue, and Claire Cardie. 2010. Multi-level structured models for document-level sentiment classification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1046–1056. Association for Computational Linguistics.

Zhengchen Zhang, Minghui Dong, and Shuzhi Sam Ge. 2014. Emotion analysis of children's stories with context information. In *Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA)*, pages 1–7. IEEE.

Ge S S Zhang Z, Dong M. 2014. Emotion analysis of children's stories with context information. In *Asia-Pacific Signal and Information Processing Association*, volume 38, pages 1–7. IEEE.

Jun Zhao, Kang Liu, and Gen Wang. 2008. Adding redundant features for crfs-based sentence sentiment classification. In *Proceedings of the conference on empirical methods in natural language processing*, pages 117–126. Association for Computational Linguistics.