School of Informatics

Informatics Research Review
Sarcasm Detection in English Tweets:
Dataset and Feature Creation Approaches

[redacted]

January 2021

Abstract

Due to its polarity flipping nature, sarcasm detection in textual data is a challenging, yet important task in Natural Language Processing. This review discusses sarcasm detection in English tweets using the binary classification approach. Specifically, we studied dataset creation approaches and feature creation approaches across literature. We found that two common dataset creation approaches are hashtag-based annotation and manual annotation. Moreover, these approaches represent different sarcasm phenomena. Related to sarcasm context, we found two different feature types: local features and contextual features. Finally, studies show that sarcasm detection systems perform best when both local features and contextual features are used.

Date: Thursday 14th January, 2021

Supervisor: [redacted]

1 Introduction

Sarcasm is a type of figurative language where the speakers state the opposite of what they mean. Formally, the online Cambridge English dictionary defines sarcasm as "the use of remarks that clearly mean the opposite of what they say, made in order to hurt someone's feelings or to criticize something in a humorous way". Because the real meaning cannot be derived directly from the sentence, sarcasm detection is a challenging problem in Natural Language Processing. The difficulty is even more exacerbated when sarcasm detection is performed in textual data because there are no tonal, gestural, or facial expressions that can be used to interpret the meaning of the utterance. This problem suggests that sarcasm is a contextual phenomenon because the exact same sentence can be sarcastic or not sarcastic, depending on the context in which it is conveyed. For example, the sentence "*Thank you for being on time.*" is sarcastic when it is said to a late person. However, the sentence is not sarcastic if the person is really on time. The study of Gerrig and Goldvarg [1] provides empirical evidence of the importance of situational context in the perception of sarcasm.

Although difficult, sarcasm detection in textual data remains an important task in Natural Language Processing. It is the first step to correctly interpreting the meaning of a text, which is important for some domains, such as chatbots [2, 3], customer reviews [4, 5] or sentiment analysis [6, 7]. For instance, the study by Bouazizi and Ohtsuki [7] shows that the performance of sentiment detection systems can be improved when the systems are aware of sarcastic nature of the text. Therefore, over the past decade, much research has explored sarcasm detection in textual data, such as product reviews [4, 5], online discussions [8, 9], and tweets [10, 11, 12, 13].

This research review discusses automatic sarcasm detection in English tweets using the binary classification approach. More specifically, we focus on the dataset creation and feature creation parts of the classification task. We limit our discussion to this scope because of several reasons. Firstly, English tweets are widely used in sarcasm detection studies due to the large number of English Twitter users and the ease of collecting Twitter data via Twitter API. Secondly, binary classification is the most common approach used for sarcasm detection in literature, where a text is categorized only as sarcastic or non-sarcastic [14]. Lastly, dataset creation and feature creation are the initial, yet essential parts of a classification task. Hence, we will not discuss sarcasm detection work which used other sources of textual data (e.g. product reviews [4, 5], online discussions [8, 9]), other languages (e.g. Dutch[15], Arabic [16]), or other detection techniques (e.g. multiclass classification [17]). In addition, some basic knowledge of machine learning and Natural Language Processing is expected from readers.

With regards to the scope and the critical role of context in sarcasm, this review will aim to answer the following research questions:

1. Are there different approaches in dataset creation for sarcasm detection? If so, do the resulting datasets represent different sarcasm phenomena?

2. In relation to contextual information, are there different approaches in feature creation for sarcasm detection? If so, what are the common approaches used in the literature?

3. Based on the answers to question 1 and 2, which feature creation approach results in a better performance on each type of the dataset?

The remaining of this paper is arranged as follows. Section 2 answers the research question 1 by discussing different approaches in dataset creation. Specifically, Subsection 2.1 reviews the hashtag-based annotation approach, and Subsection 2.2 reviews the manual annotation approach. Ultimately, Subsection 2.3 answers the question of whether the resulting datasets represent different sarcasm phenomena. Next, Section 3 answers the research question 2 by reviewing feature creation approaches in literature. Subsection 3.1 reviews the approach which uses only information in the target text (local features). Following that, Subsection 3.2 reviews the approach which leverages information beyond the target text (contextual features), and concludes by answering the research question 3 based on the whole discussion in Section 3. Finally, Section 4 provides a summary and a conclusion.

2 Dataset Creation

In general, dataset creation for classification task involves data gathering and data annotation. Research which uses Twitter data can gather tweets via Twitter API. Moreover, it is possible to obtain tweets with specific hashtags by specifying them as filtering criteria [18]. Common hashtags used in sarcastic tweets are #sarcasm, #sarcastic, and #not [11, 19, 20, 21].

In terms of data annotation, there are two common ways to obtain labels for sarcasm detection in literature. The first approach is to derive the label automatically from the presence of sarcasm-related hashtags in the tweet (hashtag-based annotation) [19, 20, 21]. The second approach is to manually label the dataset (manual annotation) [11, 12, 22].

2.1 Hashtag-Based Annotation

In hashtag-based annotation, tweets are considered positive examples if they contain sarcasm-related hashtags, such as #sarcasm, #sarcastic, or #not [19, 20, 21]. On the other hand, there are several ways to select negative samples for the dataset. First, negative examples can be obtained from random tweets without sarcasm-related hashtags [20, 21]. Second, negative samples can be tweets with specific hashtags defined by the researchers, such as #notsarcasm, #notsarcastic [23]; #education #humor, #politics [24]; or #happy, #lucky, #sadness, #frustated [10]. An example of this type of dataset is the Ptacek dataset [19]. It consists of 50,000 sarcastic tweets and 50,000 non-sarcastic tweets. The sarcastic tweets were obtained using the #sarcastic hashtag. Nonetheless, the author did not specify the criteria for the non-sarcastic tweets.

There are several advantages of this automatic approach. The main advantage is it allows creating large-scaled labelled datasets quickly. For example, the size of manually-annotated datasets in some studies [11, 12, 22, 25] is around 2000-5000 instances, whereas the size of automatically-annotated datasets in other studies [24, 19] can reach around 60,000-100,000 instances . This approach also requires a relatively low cost since it involves minimal human resources. In addition, Gonzalez-Ibanez et al. [10] argue that the best judge of whether a tweet is sarcastic or not is the author of the tweet.

However, this approach also has several disadvantages. First, it might introduce bias to the datasets by capturing tweets only of certain types or from certain users. Bamman and Smith [21] found that the #sarcasm hashtag is commonly used in cases where the author wants to prevent the audience from drawing an incorrect conclusion about the tweet. For instance, “celebrity” users who are anxious that the public would misinterpret their messages. Second, the labels are often noisy, i.e. positive labels can be incorrectly assigned to certain tweets. Gonzalez-Ibanez et al. [10] discovered that some tweets with the #sarcasm hashtag were not sarcastic, but rather about the sarcasm itself. Usually, this kind of tweet has the #sarcasm hashtag in the middle of the tweet, e.g. “*It was #sarcasm when I suggested you to keep going*”. These tweets can lead to false positives. In addition, Riloff et al. [11] noted that there might be sarcastic tweets that do not contain sarcasm-related hashtags, which can lead to false negatives.

2.2 Manual Annotation

An alternative to hashtag-based annotation is asking human annotators to identify whether a tweet is sarcastic or not. The tweets can be collected in a completely random fashion [22], or in a way that some collected tweets contain the #sarcasm hashtags [11]. Nonetheless, all hashtags will be removed before presenting the tweets to the annotators. The Riloff dataset [11] is an example of a manual dataset which is widely used in literature [13, 26, 27]. It consists of 3,200 tweets comprising 742 sarcastic tweets and 2,458 non-sarcastic tweets.

The manual annotation has the advantages of reducing label noise in the dataset, which can lead to false positives or false negatives. Gonzalez-Ibanez et al. [10] found that even after collecting only tweets with the #sarcasm hashtag located at the end of the message (the position where people usually put the tag to signal that the tweet is sarcastic), there were still non-sarcastic tweets captured. For instance, the tweet "*I love #sarcasm*" does not express sarcasm. Without the manual review, the tweet would be labelled as positive. In addition, this approach can also capture sarcastic tweets which do not contain any sarcasm-related hashtags. For example, "*It is just great that my iPhone was stolen*" [11].

Nevertheless, this approach has also several drawbacks. First, it requires a large amount of time and money due to human workers involvement. Second, there might be subjective bias in the labelling process because of the different perception about sarcasm between annotators. Hence, it is crucial to measure the consistency between annotators, and thus the reliability of the labels by calculating the inter-annotator agreement metric, such as Fleiss' κ [4] or Cohen's κ [11]. Even when the inter-annotator agreement is high as reported in [11] (Cohen's κ values were around 0.8), it is important to question whether the annotators are sufficiently diverse and representative of the audience of the tweets. This scepticism is based on the finding by Joshi et al. [28] which suggests that annotators from different socio-cultural backgrounds, e.g. Indian and American, tend to have higher disagreement than annotators from the same nationality.

2.3 Dataset Creation Approaches and The Reflected Sarcasm Phenomena

Discussions in Section 2.1 and 2.2 suggest that both hashtag-based annotation and manual annotation have their own advantages and disadvantages. However, the finding by Oprea and Magdy [27] suggests that deciding which approach to use is more than a matter of trade-off decision-making.

Oprea and Magdy [27] investigated the impact of using author context as features in sarcasm detection. They defined author context as the embedded representation of the author's historical tweets, which was extracted using neural networks. This embedded representation was referred to as *user embedding*. In one of the experiment settings, user embeddings were used as the only features for predicting sarcasm in tweets. The model which used user embeddings as the only features was referred to as *exclusive model*. Moreover, the authors experimented on two datasets which were labelled using different annotation approaches: the Riloff dataset [11] (labelled manually) and the Ptacek dataset [19] (labelled automatically).

Experiments with the exclusive model indicated that user embeddings had a significant predictive power on the Ptacek dataset, whereas they are far less informative on the Riloff dataset. Compared to the baseline model adopted from Tay et al. [26], the exclusive model achieved a larger F-score on the Ptacek dataset (0.922 v.s. 0.863), but not on the Riloff dataset (0.546 v.s 0.711). Oprea and Magdy argued that this performance discrepancy was rooted in the two datasets, which represented different sarcasm phenomena. The manually labelled dataset (the Riloff dataset) represented perceived sarcasm (sarcasm from the audience's perspective), whereas the automatically labelled dataset (the Ptacek dataset) represented intended sarcasm (sarcasm from the author's perspective). Therefore, it is not surprising if user embeddings were more predictive of sarcasm as intended by the author. The authors conclude that the two annotation approaches capture different sarcasm phenomena and suggest future research to take into account this difference in their work.

Presumably, previous research was not aware of the different sarcasm phenomena reflected by the two different dataset creation approaches. When creating the Riloff dataset, Riloff et al. [11] collected 1,600 tweets containing sarcasm-related hashtags and 1,600 random tweets without those hashtags. After that, they removed all hashtags and asked human annotators to judge whether a tweet was sarcastic or not. Surprisingly, only 713 of the 1,600 tweets with sarcasm-related hashtags were judged to be sarcastic. This finding should have indicated that there was a difference between intended sarcasm and perceived sarcasm. However, Riloff et al. failed to recognise this fundamental difference. On the contrary, they attributed the lack of coherence between manual annotations and the presence of sarcasm-related hashtags to the lack of conversation context. Therefore, the finding by Oprea and Magdy serves as an important wake-up call to researchers in sarcasm detection to acknowledge the difference between the two types of sarcasm and decide on the annotation approaches used in their research accordingly.

3 Feature Creation

As described in Section 1, context plays an important role in understanding sarcasm. With regards to the use of context, there are two common approaches in feature creation for sarcasm detection: the local feature approach [10, 13] and the contextual feature approach [27, 29].

The local feature approach extract features only from the tweet content itself. Several types of local features are lexical features, pragmatic features, and pattern-based features [30]. Lexical features are linguistic properties derived from the text, e.g. unigrams, punctuations, and part-of-speech tags [31]. In contrast, pragmatic features are features other than the words in the text which are used to clarify the intention of the message and compensate for the absence of nonverbal clues in written communication, e.g. emoticons, emojis [32, 33]. Finally, pattern-based features are features which seek the presence of some patterns in the text, such as a positive sentiment word in a negative context [11].

In contrast, the contextual feature approach extract features from the information beyond the target text (contextual information). Several types of contextual information are author-specific context and conversation context [29]. The author-specific context refers to any information specific to the author of the tweet (e.g. user profile and historical tweets), whereas conversation context refers to other tweets in the conversation of which the target tweet is involved.

3.1 Local Features

The study by Gonzalez-Ibanez et al.(2011) [10] is one of the earliest work in sarcasm detection using Twitter data. This study investigated the impact of lexical and pragmatic features on machine learning performance for identifying sarcastic tweets. The authors used the hashtag-based annotation approach for building the dataset by collecting 900 sarcastic tweets containing

#sarcasm or #sarcastic hashtags, and 1,800 non-sarcastic tweets containing positive hashtags (#joy, #happy, #lucky) or negative hashtags (#angry, #sadness, #frustrated).

The authors experimented with local features, specifically lexical features and pragmatic features. Two kinds of lexical features used were unigrams and dictionary-based features. The dictionary-based features were in the form of presence or frequency of word categories [34], affective words [35], interjections, and punctuations. In contrast, the pragmatic features were the presence and frequency of positive emoticons (e.g. smileys), negative emoticons (e.g. frowning faces), and *@user* which signalled that the tweet was a reply to another tweet.

The experiments using SVM with sequential minimal optimization (SMO), Logistic Regression, and several feature combinations achieved only an average accuracy of 62%. The best accuracy of 65.44% was obtained using SMO and unigrams as features. Based on these results, the authors concluded that the lexical and pragmatic features used in this study were not sufficient for accurately identifying sarcastic tweets, especially because the features were derived from isolated tweets without any contextual information. Hence, Gonzalez-Ibanez et al. suggested future work to investigate the impact of using contextual features on sarcasm detection in tweets. Another weakness that we noticed is the results were reported in accuracy, whereas the dataset is imbalanced, i.e. the positive to negative ratio is 1:2. Hence, the seemingly good result (65.44%) could have been obtained from correctly predicting only the negative samples. Therefore, the authors should have used another metric which measures a system performance more objectively on an imbalanced dataset, such as F-score.

In addition to lexical features, Riloff et al.(2013) [11] proposed the use of pattern-based features to detect sarcasm based on sentiment incongruity. More specifically, Riloff et al. argued that a sarcastic tweet could be identified by the presence of a positive sentiment phrase in a negative context. For instance, in the remark of "*Absolutely adore it when my bus is late*", the positive sentiment word "*adore*" is in close proximity to a negative situation phrase "*my bus is late*". This way of detecting sarcasm was referred to as the Contrast method. The dataset used was manually annotated and composed of 742 sarcastic tweets and 2,458 non-sarcastic tweets. Experimental results showed that an SVM classifier with lexical features (unigrams and bigrams) could detect sarcastic tweets with an F-score of 0.48. However, when the labels from the SVM classifier were combined with the labels from the Contrast method, the resulting F-score increased to 0.51. These results suggest that pattern-based features can improve the performance of an automatic sarcasm detection system. Nonetheless, Riloff et al. acknowledged that the pattern used in this study was still highly constrained. Therefore, they suggested future work to explore methods which could learn more flexible patterns of sentiment incongruity. Moreover, in contrast to the pattern explored in their study, Riloff et al. pointed out that sarcasm might also be in the form of a negative sentiment phrase followed by a positive situation, e.g. "*Going to the dentist for a root canal this afternoon. Yay, I can't wait*". Hence, the authors also suggested future work to explore this different pattern for detecting sarcasm.

In response to the suggestion from Riloff et al. [11], Bharti et al. (2015) [12] extended the pattern-based feature used in [11] with the other type of sentiment incongruity: the presence of negative sentiment in a positive context. An example for this type of sarcasm is "*I hate Australia in cricket because they always win*". In this sentence, the negative sentiment word "*hate*" contrasts with the positive situation "*always win*". In addition, Bharti et al. also investigated another pattern-based feature to identify sarcasm: the presence of interjection words (e.g. *wow, yay*) at the beginning of a tweet. This feature was based on the assumption that sarcastic tweets are often started with an interjection word, for instance, "*Wow, now that I know about Photoshop, my decade-long eating disorder is cured!*". For experiments, the authors created a manually annotated dataset for each of the pattern-based features. The first dataset was for the sentiment incongruity feature and consisted of 3,000 random tweets. The second dataset was for the interjection feature and consisted of 3500 tweets, all of which started with an interjection word. Next, pattern matching algorithms were used to label a tweet as sarcastic if it contained any of the pattern-based features. Using the sentiment incongruity feature, the algorithm achieved an average F-score of 0.765 on the first dataset. Using the interjection feature, the algorithm achieved an average F-score of 0.82 on the second dataset. These results further support that pattern-based features are useful for detecting sarcasm in tweets. However, the robustness of the interjection feature still needs further proof because in this study, it was only tested on a dataset containing tweets which were all started with interjection words.

So far, the earlier sarcasm detection work discussed in this section still used hand-crafted features for extracting information from the tweets. With the advances in deep learning, more recent work of sarcasm detection started to employ deep learning for extracting local features from the tweets. One of the earliest studies which applied deep learning for sarcasm detection is the study by Ghosh and Veale (2016) [13]. They attempted to detect sarcasm in tweets by modelling the semantic of sentences using a neural network model. More precisely, the tweets were fed into a model composed of a word embedding layer, a CNN, an LSTM, and a fully connected deep neural network. The proposed method successfully achieved an F-score of 0.881 on the Riloff dataset [11], significantly outperforming the original work by Riloff et al. (F-score: 0.51). This result suggests that deep learning is a promising approach in sarcasm detection.

As sarcasm is often associated with a contrastive relationship between sentiments in a text, Tay et al. (2018) [26] criticised the approach of Ghosh and Veale [13] which did not take into account interaction between words in the tweets by merely feeding tweet texts into a deep neural model. Therefore, Tay et al. proposed an attention-based deep neural model which could explicitly model the semantics between each word in the tweets. The authors claimed that this model was the first attention model for sarcasm detection. Moreover, this work can be regarded as a response to a suggestion from Riloff et al. [11] for a more flexible method to detect sentiment incongruity between words in a text because the incongruity was learned automatically by the deep learning model.

The proposed model learned two representations from a text: an intra-attentive representation and a compositional representation. The intra-attentive representation held the information about relationship between words in the text because it was learned by concatenating each word embedding pair in the text and feeding them into a sequence of dense layers and attention layers. This intra-attentive representation only considered word-by-word interaction and could not capture the semantic of a compositional phrases, such as "*not happy*". Therefore, the proposed model learned the second type of representation, i.e. the compositional representation, by encoding the whole text using an LSTM model. The two representations were then concatenated and fed into a dense layer for the final prediction.

For evaluation, the authors compared the performance of their proposed model with the model of Ghosh and Veale [13] on the Riloff dataset [11] and the Ptacek dataset [19]. Experimental results showed that the proposed model outperformed Ghosh and Veale's model on both dataset. The proposed model achieved F-scores of 0.7324 and 0.8600 on the Riloff dataset and the Ptacek dataset respectively, whereas Ghosh and Veale's model achieved only 0.6781 and 0.8374. Note that the F-score of Ghosh and Veale's model on the Riloff dataset reported in this paper (0.6781) was different from the one reported in the original paper (0.881) [13]. It is likely because in this paper, the authors only used 1,954 of the 3,200 tweets in the original dataset. Although the authors did not explain about this discrepancy, a possible explanation is that they did not manage to collect all original tweets via Twitter API since some tweets might have been deleted. With these results, the attention model proposed by Tay et al. successfully achieved

Feature Class	Feature Type	Feature Examples
Tweet Features	Local	unigrams, bigrams, Brown cluster unigrams & bigrams [37], part of speech features, pronunciation features, capitalization features, tweet whole sentiment, tweet word sentiment, intensifiers.
Author Features	Contextual	author historical salient terms, historical topics, historical sentiment, profile information, profile unigrams.
Audience Features	Contextual	addressee historical salient terms, historical topics, profile information, profile unigrams, author/addressee interactional topics, author/addressee historical communication.
Response Features	Contextual	pairwise Brown features between the original message and the response, unigram features of the original message.

Table 1: Features in Bamman and Smith (2015) [21]

the state-of-the-art performance on the Riloff dataset and the Ptacek dataset in 2018.

As discussed in the Section 1, contextual information plays an essential role in identifying sarcasm. Moreover, some studies on sarcasm detection which used local features have also acknowledged the importance of contextual information for sarcasm detection [11, 10, 23]. This importance is further confirmed by the work of Wallace et al. [36] which provides empirical evidence that human annotators often require context to judge whether a content is sarcastic or not. Moreover, machine learning approaches also tend to misclassify the same instances for which the annotators requested additional context. Although the experiment was conducted on a Reddit dataset, the findings provide insights into sarcastic text detection in general. Thus, much research in sarcasm detection have tried to incorporate contextual information as features in their model. In the next section, we will discuss some studies which explored this direction.

3.2 Contextual Feature

One of the earliest studies which leveraged contextual features is the study by Bamman and Smith (2015) [21]. This study criticised previous work in sarcasm detection which used only linguistic features from the tweet text (local features), and aimed to prove that the use of extra-linguistic information from the tweet context could bring performance improvement.

The authors proposed four classes of features, which encompassed various hand-crafted local and contextual features. The four feature classes were: Tweet Features, Author Features, Audience Features, and Response Features. The Tweet Features were local features whose scope only over the target tweet. The Author Features were contextual features related to the author of the target tweet (author-specific context). The Audience Features were contextual features related to the person to whom the target tweet responded, including the author's historical interaction with them. Finally, the Response Features were contextual features about the interaction between the target tweet and the tweet that it was responding to (conversation context). Table 1 shows the examples for each of the feature classes.

In the experiments, the authors investigated the performance of an l2-regularized binary logistic regression model on an automatically annotated dataset using various feature class combinations. The dataset was balanced and consisted of 9,767 sarcastic samples and 9,767 non-sarcastic tweets. For each tweet author in the dataset, up to 3,200 most recent tweets were collected for contextual information. Experimental results showed that the addition of any contextual feature classes (Author, Audience, Response) yielded improvement over the Tweet Features. The greatest improvement (75.4% to 84.9%) resulted from the addition of the Author Features to the Tweet Features. In addition, the author historical salient terms (i.e. 100 historical terms with the highest TF-IDF score) were found to be the most informative features. In conclusion, this study empirically proved that contextual information is useful for detecting sarcasm in tweets.

Inspired by Ghosh and Veale's success [13] in adopting deep learning for sarcasm detection using local features, Zhang et al. (2016) [29] conducted a study to apply deep learning for sarcasm detection with contextual information. The dataset used was annotated using hashtag-based annotation. It consisted of 9,104 positive samples and 9,104 negative samples. In addition, 80 past tweets from each tweet author were collected for the contextual features.

The authors conducted performance comparison between a traditional model with manual features and a deep learning model. Both models were composed of a local component and a contextual component. The local component was designed to extract local features from the target tweet, whereas the contextual component was used to extract contextual features from the author's historical tweets. For the traditional model, the local features were unigrams, bigrams, and trigrams. In addition, the contextual features were the author's historical salient terms, which was adopted from the work of Bamman and Smith [21]. The deep learning model was designed in such a way that the feature sources were comparable with those of the traditional model. The local component of the deep learning model was a bi-directional gated recurrent neural network (GRNN) whose input was a concatenation of three consecutive word vectors $\{w_{t-1}, w_t, w_{t+1}\}$. This input was analogous to the trigrams in the traditional model. Moreover, the contextual component of the deep learning model was a gated pooling function applied to the same contextual tweet words used in the traditional model. Experimental results showed that the deep learning model achieved higher accuracy and F-score compared to the traditional model. The accuracy and F-score for the deep learning model were both 90.74, whereas the accuracy and F-score for the traditional model were 79.29 and 79.36, respectively. Hence, this study complement the study by Ghosh and Veale [13] by showing that deep learning approaches can outperform traditional approaches not only when used with local features, but also when used with contextual features.

Following the success of Zhang et al. [29] with the deep learning approach, Ghosh et al. (2018) [38] investigated the impact of conversation context for sarcasm detection using a deep learning approach. The dataset was annotated using hashtag-based annotation and consisted of 12,215 sarcastic tweets and 13,776 non-sarcastic tweets. The length of the conversation context was varied with 30% of the tweets had more than one tweet in the context. The authors experimented with various architectures of LSTM and the baseline model was a simple LSTM which read only the target tweet. Results showed that the best model for identifying sarcastic tweets was the one which encoded the target tweet and the context tweets separately using different attention-based LSTMs, before concatenating the two embeddings for the final prediction. This best model achieved a significantly larger F-score (0.737) than the baseline model (0.671). Hence, this study provides empirical evidence that conversation context features contributed positively to the performance of deep learning-based sarcasm detection systems.

In contrast to the work of Ghosh et al. [38] which explored the usefulness of conversation context, Oprea and Magdy (2019) [27] investigated the usefulness of author-specific context for sarcasm detection with deep learning approach. As discussed in the Section 2.3, the exclusive model successfully outperformed the baseline model (Tay et al. [26]) on the Riloff dataset, but not on the Ptacek dataset. However, the author conducted another experiment which used both the user embedding and the target tweet embedding as features. The model that used these two embeddings as features was referred to as *inclusive model.* With the inclusion of the target tweet embedding, the inclusive model successfully outperformed the baseline model on both the Ptacek dataset (0.934 v.s. 0.863) and the Riloff dataset (0.739 v.s. 0.711). Note that the F-scores of the baseline model reported in this paper were different from the ones reported in the original paper [26] because Oprea and Magdy reimplemented the baseline model and evaluated it only on the subset of the original data. The reason was some tweet IDs were already deleted or the corresponding author accounts had been disabled. Therefore, the inclusive model successfully achieved the state-of-the-art performance on both the Riloff dataset and the Ptacek dataset in 2019, suggesting that the addition of contextual features benefited sarcasm detection on both manually-annotated dataset and automatically-annotated dataset.

In summary, research shows that automatic sarcasm detection systems achieved better performances when local features were used together with contextual features, especially when deep learning is employed [27, 38]. Oprea and Magdy [27] proved that statement using the combination of local features and author-specific context features on both manually-annotated dataset and automatically-annotated dataset. Furthermore, Ghosh et al. [38] proved that statement using the combination of local features and conversation context features on an automatically-annotated dataset. Nonetheless, to the best of our knowledge, there have not been any studies which provide empirical evidence for that statement using the combination of local features and conversation context features on a manually-annotated dataset. This work can be a suggestion for future studies. However, we can conclude that in general, the addition of contextual features bring performance improvement on both dataset types.

4 Summary & Conclusion

This review has studied dataset creation approaches and feature creation approaches for automatic sarcasm detection in English tweets. Table 2 summarises of the dataset creation and the feature creation approaches in the literature reviewed in this paper.

The two common approaches in dataset creation are the hashtag-based annotation and the manual annotation. Both approaches have their own advantages and disadvantages. However, more studies used the hashtag-based annotation approach, probably due to the convenience of obtaining a large number of data at minimal cost. In 2019, Oprea and Magdy [27] provided empirical evidence that different dataset creation approaches result in datasets representing different sarcasm phenomena. The hashtag-based annotation results in a dataset representing intended sarcasm, whereas the manual annotation results in a dataset which representing perceived sarcasm. Therefore, in addition to advantages and disadvantages, future research should also consider this difference when choosing which dataset creation approach to use.

In relation to sarcasm context, there are two approaches commonly used in literature: the local feature approach and the contextual feature approach. Earlier work in sarcasm detection tended to use the more simpler approach (local features) which extract information only from the target text [10, 11, 12]. Nevertheless, started from around 2015, more studies have explored the use of contextual features [21]. In addition, recent research has also applied deep learning in sarcasm detection [13, 26, 29, 38, 27] and enjoyed performance improvement compared to the traditional approach.

Studies also show that sarcasm detection systems achieved better performances when local features are used together with contextual features [38, 27]. In general, this finding is true for both automatically-annotated datasets and manually-annotated datasets. Although sarcasm detection systems which use only local features could also perform well (especially if deep learning is applied) [13, 26], the addition of contextual features would almost always result in a performance improvement.

Despite the advances in sarcasm detection due to the use of deep learning and contextual features, there remains some potential work and research directions in this field. First, one of the major problem in this field is the lack of standard public datasets since most researchers created their own datasets for their experiments. Although there are some datasets which are widely used across literature, such as the Riloff dataset (2013) and the Ptacek dataset (2014), these datasets are relatively old such that many of the tweet IDs or author profiles cannot be retrieved anymore [27]. This problem results in the small number of examples that can be used by more recent work and difficulties in comparing results with previous studies. Therefore, there is a need to create a new standard public dataset which allows objective performance comparison across studies. This new standard dataset should ideally provides conversation and author context of the tweets since the current trend in this field leverages contextual features. Second, with the current prevalence of deep learning application in this field, it might be useful to explore explainable deep sarcasm detection models. The insights from these explainable models could help researchers understand more about which textual or contextual components trigger sarcasm. This knowledge can then be used to further improve the performance of automatic sarcasm detection systems.

Authors	Annotation		Local Features				Contextual Features		Key Information
	Hashtag -based	Manual	Lexical	Pragmatic	Pattern	Deep Learning	Author	Conversation	
Gonzalez -Ibanez et al. (2011)	v		v	v					One of the earliest work in textual sarcasm detection. Used hand-crafted lexical and pragmatic features.
Riloff et al. (2013)		v	v		v				Proposed a manual pattern-based feature: positive sentiment in a negative context. Created the manually-annotated Riloff dataset.
Bharti et al. (2015)		v			v				Investigated another manual pattern-based feature: negative sen iment in a positive context.
Ghosh and Veale (2016)		v				v			One of the earliest work which applied deep learning for sarcasm detection. Used local features. Outperformed Riloff et al. (2013) on the Riloff dataset.
Tay et al. (2018)	v	v				v			Used a deep learning approach and local features. Outperformed Ghosh and Veale (2016) and achieved SOTA in 2018 on the Ptacek and Riloff dataset.
Bamman and Smith (2015)	v		v				v	v	One of the earliest work which incorporated manual contextual features. Showed the use of manual contextual features (either author-specific or conversation) improved the model performance.
Zhang et al. (2016)	v					v	v		One of the earliest work which applied deep learning for extracting contextual features. Used author-specific context. Showed the deep learning approach outperformed the traditional approach with manual features.
Ghosh et al. (2018)	v					v		v	Used a deep learning approach and conversation context. Showed the use of conversation context improved the model performance.
Oprea and Magdy (2019)	v	v				v	v		Used a deep learning approach and author-specific context. Outperformed Tay et al. (2018) on both the Riloff and Ptacek dataset. Showed the use of author-specific context benefited sarcasm detection on both manually-annotated and automatically-annotated dataset.

Table 2: The summary of dataset creation and feature creation approaches across literature

References

[1] Richard J Gerrig and Yevgeniya Goldvarg. Additive effects in the perception of sarcasm: Situational disparity and echoic mention. *Metaphor and Symbol*, 15(4):197–208, 2000.

[2] Florian Johannsen, Susanne Leist, Daniel Konadl, and Michael Basche. Comparison of commercial chatbot solutions for supporting customer interaction. 2018.

[3] Dimitrios Buhalis and Emily Cheng Siaw Yen. Exploring the use of chatbots in hotels: technology providers' perspective. In *Information and Communication Technologies in Tourism 2020*, pages 231–242. Springer, 2020.

[4] Dmitry Davidov, Oren Tsur, and Ari Rappoport. Semi-supervised recognition of sarcasm in twitter and amazon. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 107–116, 2010.

[5] Antonio Reyes and Paolo Rosso. Making objective decisions from subjective data: Detecting irony in customer reviews. *Decision support systems*, 53(4):754–760, 2012.

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

[7] Mondher Bouazizi and Tomoaki Ohtsuki. Opinion mining in twitter how to make use of sarcasm to enhance sentiment analysis. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pages 1594–1597, 2015.

[8] Silvio Amir, Byron C Wallace, Hao Lyu, and Paula Carvalho Mário J Silva. Modelling context with user embeddings for sarcasm detection in social media. *arXiv preprint arXiv:1607.00976*, 2016.

[9] Y Alex Kolchinski and Christopher Potts. Representing social media users for sarcasm detection. *arXiv preprint arXiv:1808.08470*, 2018.

[10] Roberto González-Ibánez, Smaranda Muresan, and Nina Wacholder. Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586, 2011.

[11] Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 704–714, 2013.

[12] Santosh Kumar Bharti, Korra Sathya Babu, and Sanjay Kumar Jena. Parsing-based sarcasm sentiment recognition in twitter data. In *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 1373–1380. IEEE, 2015.

[13] Aniruddha Ghosh and Tony Veale. Fracking sarcasm using neural network. In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 161–169, 2016.

[14] Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. A report on the 2020 sarcasm detection shared task. *arXiv preprint arXiv:2005.05814*, 2020.

[15] CC Liebrecht, FA Kunneman, and APJ van Den Bosch. The perfect solution for detecting sarcasm in tweets# not. 2013.

[16] Dana Al-Ghadhban, Eman Alnkhilan, Lamma Tatwany, and Muna Alrazgan. Arabic sarcasm detection in twitter. In *2017 International Conference on Engineering & MIS (ICEMIS)*, pages 1–7. IEEE, 2017.

[17] Zelin Wang, Zhijian Wu, Ruimin Wang, and Yafeng Ren. Twitter sarcasm detection exploiting a context-based model. In *international conference on web information systems engineering*, pages 77–91. Springer, 2015.

[18] Stream Tweets in real-time. `https://developer.twitter.com/en/docs/tutorials/stream-tweets-in-real-time`. [Online; accessed 17-December-2020].

[19] Tomáš Ptáček, Ivan Habernal, and Jun Hong. Sarcasm detection on czech and english twitter. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223, 2014.

[20] Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 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, 2015.

[21] David Bamman and Noah A Smith. Contextualized sarcasm detection on twitter. In *Ninth international AAAI conference on web and social media*. Citeseer, 2015.

[22] Gavin Abercrombie and Dirk Hovy. Putting sarcasm detection into context: The effects of class imbalance and manual labelling on supervised machine classification of twitter conversations. In *Proceedings of the ACL 2016 student research workshop*, pages 107–113, 2016.

[23] Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 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: Short Papers)*, pages 757–762, 2015.

[24] Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. Modelling sarcasm in twitter, a novel approach. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 50–58, 2014.

[25] Cynthia Van Hee, Els Lefever, and Véronique Hoste. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, 2018.

[26] Yi Tay, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. Reasoning with sarcasm by reading in-between. *arXiv preprint arXiv:1805.02856*, 2018.

[27] Silviu Oprea and Walid Magdy. Exploring author context for detecting intended vs perceived sarcasm. *arXiv preprint arXiv:1910.11932*, 2019.

[28] Aditya Joshi, Pushpak Bhattacharyya, Mark Carman, Jaya Saraswati, and Rajita Shukla. How do cultural differences impact the quality of sarcasm annotation?: A case study of indian annotators and american text. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 95–99, 2016.

[29] Meishan Zhang, Yue Zhang, and Guohong Fu. Tweet sarcasm detection using deep neural network. In *Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers*, pages 2449–2460, 2016.

[30] Muhammad Abulaish, Ashraf Kamal, and Mohammed J Zaki. A survey of figurative language and its computational detection in online social networks. *ACM Transactions on the Web (TWEB)*, 14(1):1–52, 2020.

[31] Roger Kreuz and Gina Caucci. Lexical influences on the perception of sarcasm. In *Proceedings of the Workshop on computational approaches to Figurative Language*, pages 1–4, 2007.

[32] Dominic Thompson and Ruth Filik. Sarcasm in written communication: Emoticons are efficient markers of intention. *Journal of Computer-Mediated Communication*, 21(2):105–120, 2016.

[33] Li Li and Yue Yang. Pragmatic functions of emoji in internet-based communication—a corpus-based study. *Asian-Pacific Journal of Second and Foreign Language Education*, 3(1):16, 2018.

[34] James W Pennebaker, Roger J Booth, and Martha E Francis. Linguistic inquiry and word count: Liwc [computer software]. *Austin, TX: liwc. net*, 135, 2007.

[35] Carlo Strapparava, Alessandro Valitutti, et al. Wordnet affect: an affective extension of wordnet. In *Lrec*, volume 4, page 40. Citeseer, 2004.

[36] Byron C Wallace, Laura Kertz, Eugene Charniak, et al. Humans require context to infer ironic intent (so computers probably do, too). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 512–516, 2014.

[37] Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 380–390, 2013.

[38] Debanjan Ghosh, Alexander R Fabbri, and Smaranda Muresan. Sarcasm analysis using conversation context. *Computational Linguistics*, 44(4):755–792, 2018.