mathematics

MDPI

Article

Intermediate-Task Transfer Learning with BERT for Sarcasm Detection

Edoardo Savini and Cornelia Caragea *

Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607, USA; edoardosavini95@gmail.com
* Correspondence: cornelia@uic.edu

Abstract: Sarcasm detection plays an important role in natural language processing as it can impact the performance of many applications, including sentiment analysis, opinion mining, and stance detection. Despite substantial progress on sarcasm detection, the research results are scattered across datasets and studies. In this paper, we survey the current state-of-the-art and present strong baselines for sarcasm detection based on BERT pre-trained language models. We further improve our BERT models by fine-tuning them on related intermediate tasks before fine-tuning them on our target task. Specifically, relying on the correlation between sarcasm and (implied negative) sentiment and emotions, we explore a transfer learning framework that uses sentiment classification and emotion detection as individual intermediate tasks to infuse knowledge into the target task of sarcasm detection. Experimental results on three datasets that have different characteristics show that the BERT-based models outperform many previous models.

Keywords: sarcasm detection; intermediate-task transfer learning; emotion-enriched sarcasm detection

MSC: 68T50

Citation: Savini, E.; Caragea, C. Intermediate-Task Transfer Learning with BERT for Sarcasm Detection. *Mathematics* **2022**, *10*, 844. https://doi.org/10.3390/math10050844

Academic Editor: Victor Mitrana

Received: 1 February 2022
Accepted: 1 March 2022
Published: 7 March 2022

1. Introduction

In recent years, the Internet has become the main source to communicate and share information. In particular, social media sites, microblogs, discussion forums, and online reviews have become more and more popular. They represent a way for people to express their own opinion with no inhibition and to search for some advice on various products or even vacation tips. Many companies take advantage of these sites' popularity to share their products and services, provide assistance, and understand costumer needs. For this reason, social media websites have developed into one of the main domains for the Natural Language Processing (NLP) research, especially in the areas of Sentiment Analysis and Opinion Mining. Analyzing people's sentiments and opinions could be useful to comprehend their behavior, monitor customer satisfaction, and increase sales revenue. However, these tasks appear to be very challenging [1,2] due to the dense presence of figurative languages in social media communities, such as Reddit or Twitter.

Our research focuses on a recurrent sophisticated linguistic phenomenon (and a form of speech act) that makes use of figurative language to implicitly convey contempt through the incongruity [3] between text and context: the sarcasm. Its highly figurative nature has caused sarcasm to be identified as one of the most challenging tasks in natural language processing [4], and has attracted significant attention in recent years along two lines of research: (1) understanding sarcasm from different online platforms by creating novel datasets [5–10]; and (2) designing approaches to effectively detect sarcasm from textual data. Although many previous works on this task focused on approaches based on feature engineering and standard classifiers such as Support Vector Machines to extract lexical cues recurrent in sarcasm [6,11,12], more recent works [13–15] have started to explore deep neural networks for sarcasm detection in order to capture the hidden intricacies from text.

Still, despite substantial progress on sarcasm detection, the research results are scattered across datasets and studies.

In this paper, we aim to further our understanding of what works best across several textual datasets for our target task: sarcasm detection. To this end, we present strong baselines based on BERT pre-trained language models [16]. We further propose to improve our BERT models by fine-tuning them on related intermediate tasks before fine-tuning them on our target task so that inductive bias is incorporated from related tasks [17]. We study the performance of our BERT models on three datasets of different sizes and characteristics, collected from the Internet Argument Corpus (IAC) [11], Reddit [18], and Twitter [7]. Table 1 shows examples of sarcastic comments from each of the three datasets. As we can see from the table, the dataset constructed by Oraby et al. [11] contains long comments, while the other two datasets have comments with fairly short lengths. Our purpose is to analyze the effectiveness of BERT and intermediate-task transfer learning with BERT on the sarcasm detection task and find a neural framework able to accurately predict sarcasm in many types of social platforms, from discussion forums to microblogs.

Table 1. Examples of sarcastic comments from our datasets.

Oraby et al. [11]:	*"And, let's see, when did the job loss actually start?, Oh yes.. We can trace the troubles starting in 2007, with a big melt down in August/September of 2008. Let's see.. Obama must have been a terrible president to have caused that.. oh WAIT. That wasn't Obama, that was BUSH.. Excuse Me."*
Khodak et al. [18]:	*"Obama is in league with ISIS, he wins the shittiest terrorist fighter award."*
Mishra et al. [7]:	*"I can't even wait to go sit at this meeting at the highschool."*

Our contributions are summarized as follows:

- We show that sarcasm detection results are scattered across multiple papers, which makes it difficult to assess the advancements and current state-of-the-art for this task.
- We establish strong baselines based on BERT pre-trained language models for this task. Our analysis is based on experimental results performed on three sarcasm datasets of different sizes (from small to large datasets) and covering different characteristics captured from various social platforms (from the Internet Argument Corpus to Reddit and Twitter).
- Inspired from existing research on sarcasm [6] which shows its correlation with sentiment and emotions, we find that the performance of BERT can be further improved by fine-tuning on data-rich intermediate tasks, before fine-tuning the BERT models on our sarcasm detection target task. We use diverse intermediate tasks (fine-grained emotion detection from general tweets, coarse-grained sentiment polarity by polarizing the emotions in the above dataset into positive and negative sentiment, and sentiment classification of movie reviews). We show that, depending on the characteristics of the target task data, different intermediate tasks are more useful than others. We make our code available to further research in this area (https://github.com/edosavini/TransferBertSarcasm, accessed on 23 March 2021).

2. Related Work

Experiments on automatic sarcasm detection represent a recent field of study. The first investigations made on text were focused on discovering lexical indicators and syntactic cues that could be used as features for sarcasm detection [6,11]. In fact, at the beginning, sarcasm recognition was considered as a simple text classification task. Many studies focused on recognizing interjections, punctuation symbols, intensifiers, hyperboles [19], emoticons [20], exclamations [21], and hashtags [22] in sarcastic comments. More recently, Wallace et al. [4] showed that many classifiers fail when dealing with sentences where context is needed. Therefore, newer works studied also parental comments or historical tweets of the writer [3,23,24].

In order to detect semantic and contextual information from a sarcastic statement, researchers started to explore deep learning techniques. The advantage of adopting neural networks is in their ability to induce features automatically, allowing them to capture long-range and subtle semantic characteristics that are hard to capture with manual feature engineering. For example, Joshi et al. [15] proposed different kinds of word embeddings (Word2Vec, GloVe, LSA), augmented with other features on word vector-based similarity, to apprehend context in phrases with no sentiment words. Poria et al. [25] developed a framework based on pre-trained CNNs to retrieve sentiment, emotion and personality features for sarcasm recognition. Zhang et al. [26] created a bi-directional gated recurrent neural network with a pooling mechanism to automatically detect content features from tweets and context information from history tweets. Ghosh and Veale [14] proposed a concatenation of 2-layer Convolutional Neural Networks with 2-layer Long-Short Term Memory Networks followed by a fully connected deep neural network and showed improved results over text based engineered features. Oprea and Magdy [9] studied intended vs. perceived sarcasm using CNN and RNN-based models.

Other authors leveraged user information in addition to the source text. For example, Amir et al. [13] used Convolutional Neural Networks (CNNs) to capture user embeddings and utterance-based features. They managed to discover homophily scanning a user's historical tweets. Hazarika et al. [27] proposed a framework able to detect contextual information with user embedding created through user profiling and discourse modeling from comments on Reddit. Their model achieves state-of-the-art results in one of the datasets (SARC) [8] we consider in our experiments.

Majumder et al. [28] used a Gated Recurrent Unit (GRU) with an attention mechanism within a multitask learning framework with sarcasm detection as the main task and sentiment classification as an auxiliary task and applied it on the dataset by Mishra et al. [7], which contains about a thousand tweets labeled with both sarcastic and sentiment labels. Their mechanism takes as input Glove word embeddings, shares the GRU model between the two tasks, and exploits a neural tensor network to fuse sarcasm and sentiment-specific word vectors. The authors were able to outperform the state-of-the-art previously obtained with a CNN model by Mishra et al. [29]. Plepi and Flek [30] used a graph attention network (GAT) over users and tweets from a conversation thread to detect sarcasm and used a BERT model as a baseline. Other works [31–33] focused on multi-modal sarcasm detection by analyzing the relationship between the text and images using models such as BERT [16], ResNet [34], or VilBERT [35].

In contrast to the above works, we explore BERT pre-trained language models and intermediate-task transfer learning with BERT focusing solely on the text of each user post and establish strong baselines for sarcasm detection across several social platforms.

3. Baseline Modeling

3.1. BERT Pre-Trained Language Model

The BERT pre-trained language model [16] has pushed performance boundaries on many natural language understanding tasks. We fine-tune BERT `bert-base-uncased` from the HuggingFace Transformers library [36] on our target task, i.e., sarcasm detection, with an added single linear layer on top as a sentence classifier that uses the final hidden state corresponding to the `[CLS]` token.

3.2. Intermediate-Task Transfer Learning

Several works proposed to further improve pre-trained models by first fine-tuning a pre-trained model, e.g., BERT, on an intermediate task, before fine-tuning it again on the target task [17,37]. However, these works showed that this approach does not always boost the performance of a target task. Inspired by this idea and the progress on sarcasm detection, which showed a strong correlation between sarcasm and (implied negative) sentiment and emotions [6], we propose to explore transfer learning from the related intermediate tasks of sentiment classification and emotion detection, to understand if we

can further improve the performance of our BERT models on the sarcasm detection target task. Figure 1 shows the steps taken in this transfer learning framework.

Next, we discuss our target task and the intermediate tasks used for transfer learning.

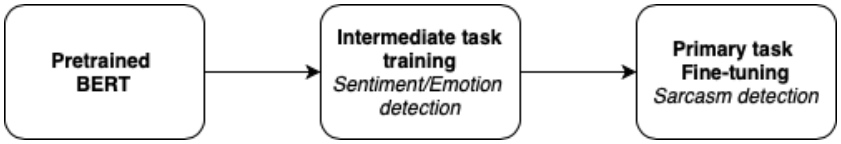

Figure 1. Our transfer learning framework.

3.2.1. Target Task

Sarcasm Detection. Our target task is sarcasm detection from textual inputs. Specifically, given a piece of text, e.g., a message, a tweet, a comment, or a sentence, the task is to predict if the text is sarcastic or not, *solely* from the text.

3.2.2. Intermediate Tasks

Fine-Grained EmoNet. EmoNet [38] is a Twitter dataset composed of tweets automatically annotated using distant supervision with the Plutchik-24 emotion set. Thus, by construction, the tweets in this dataset contain more explicit emotion-rich words. We obtained a smaller version of the dataset from the authors. This version contains about 50,000 tweets annotated with the Plutchik-8 emotion set (joy, surprise, trust, anticipation, sadness, fear, anger, disgust). We fine-tuned BERT on the EmoNet tweets in a supervised fashion before fine-tuning it on our sarcasm detection target task.

Coarse-Grained EmoNet. This dataset is the same as the EmoNet dataset above [38] except that we make the labels for each tweet more abstract according to the polarized emotions (positive and negative). We group all the emotion labels with negative insight (sadness, fear, anger, disgust) into a negative sentiment label (0) and group the remaining emotions (joy, surprise, trust, anticipation) into a positive class. We refer to this dataset as EmoNetSent. We fine-tuned BERT on EmoNetSent in a supervised fashion before fine-tuning it on our sarcasm detection target task.

IMDB Movie Review. The IMDB Movie Review dataset is a balanced sentiment dataset created by Maas et al. [39] for learning word vector representations to capture semantic information from text. It contains 50,000 polarized movie reviews labeled with binary sentiment classes (positive, negative). The authors avoided some preprocessing steps such as stemming and stop word removal in order to retain more indicative terms for sentiment. This dataset differs from the EmoNet dataset in terms of text (reviews) length and content. In fact, while EmoNet tweets contain short-length sentences explicitly dense of emotional charge, the IMDB dataset consists of very long phrases and sentences in which the sentiment (lexical) content appears more implicit and sparse along the sentences.

Table 2 shows examples from the datasets of our intermediate tasks, EmoNet and IMDB. We also give the number of examples in each of the intermediate tasks, EmoNet, EmoNetSent, and IMDB, in Tables 3, 4 and 5, respectively.

Table 2. Examples of sentences in the datasets of our intermediate tasks.

EmoNet	*"It's just so great to have baseball back. #happy"*	joy (1)
IMDB	*"I rented this movie primarily because it had Meg Ryan in it, and I was disappointed to see that her role is really a mere supporting one. Not only is she not on screen much, but nothing her character does is essential to the plot. Her character could be written out of the story without changing it much."*	0

Table 3. Number of tweets per emotion in the EmoNet dataset.

Emotion	Dataset	Training	Dev	Test
Joy	18,847	15,069	1884	1894
Sadness	9225	7400	932	893
Fear	6482	5198	643	641
Anger	3526	2795	346	385
Surprise	3451	2758	344	349
Disgust	2986	2362	309	315
Trust	2224	1779	233	212
Anticipation	1598	1312	142	144
Total	48,339	38,673	4833	4833

Table 4. Number of tweets per sentiment in the EmoNetSent dataset.

Sentiment	Dataset	Training	Dev	Test
Negative	22,219	17,755	2230	2234
Positive	26,120	20,918	2603	2599
Total	48,339	38,673	4833	4833

Table 5. Number of movie reviews per class in the IMDB dataset.

Class	Training	Dev	Test
Positive	20,000	2500	2500
Negative	20,000	2500	2500
Total	40,000	5000	5000

3.3. Standard Neural Model

BiLSTM: Since the Bidirectional Long Short Term Memory models perform generally well on text classification and exploit long term dependencies in text, we use these models as baselines for evaluation as well. A one-layer BiLSTM [40] with a hidden dimension of 100 is used to obtain features for each token, which are then mean pooled, followed by a fully connected layer and softmax.

CNN: Convolutional Neural Networks (CNNs) [41] also perform very well on many sentence classification tasks [41]. We note that CNN was generally used in prior works for our datasets. We used hyper-parameter settings from [41] when not available in prior work.

4. Data

To evaluate our models, we focus our attention on datasets with different characteristics, retrieved from different social media sites and having different sizes. Our first dataset is the Sarcasm V2 Corpus (https://nlds.soe.ucsc.edu/sarcasm2, accessed on 23 March 2021), created and made available by Oraby et al. [11]. Then, given the small size of this first dataset, we test our models also on a large-scale self-annotated corpus for sarcasm, SARC (http://nlp.cs.princeton.edu/SARC/, accessed on 23 March 2021), made available by Khodak et al. [18]. Last, in order to verify the efficacy of our transfer learning model on a dataset having a similar structure to the one used by our intermediate task, we selected also a dataset from Twitter (http://www.cfilt.iitb.ac.in/cognitive-nlp/, accessed on 29 March 2021), created by Mishra et al. [7]. The datasets are discussed below.

Sarcasm V2 Corpus. Sarcasm V2 is a dataset released by Oraby et al. [11]. It is a highly diverse corpus of sarcasm developed using syntactical cues and crowd-sourced annotation. It contains 4692 lines having both Quote and Response sentences from dialogue examples on political debates from the Internet Argument Corpus (IAC 2.0). The data is collected and divided into three categories: General Sarcasm (Gen, 3260 sarcastic comments and 3260 non-sarcastic comments), Rhetorical Questions (RQ, 851 rhetorical questions and 851 non-rhetorical questions) and Hyperbole (Hyp, 582 hyperboles and 582 non-hyperboles). We

use the Gen Corpus for our experiments and select only the text of the Response sentence for our sarcasm detection task.

SARC. The Self-Annotated Reddit Corpus (SARC) was introduced by Khodak et al. [18]. It contains more than a million sarcastic and non-sarcastic statements retrieved from Reddit with some contextual information, such as author details, score, and parent comment. Reddit is a social media site in which users can communicate on topic-specific discussion forums called *subreddits*, each titled by a post called *submission*. People can vote and reply to the submissions or to their comments, creating a tree-like structure. This guarantees that every comment has its "parent". The main feature of the dataset is the fact that sarcastic sentences are directly annotated by the authors themselves, through the inclusion of the marker "*/s*" in their comments. This method provides reliable and trustful data. Another important aspect is that almost every comment is made of one sentence.

As the SARC dataset has many variants (Main Balanced, Main Unbalanced, and Pol), in order to make our analyses more consistent with the Sarcasm V2 Corpus, we run our experiments only on the first version of the Main Balanced dataset, composed of an equal distribution of both sarcastic (505,413) and non-sarcastic (505,413) statements (total train size: 1,010,826). The authors also provide a balanced test set of 251,608 comments, which we use for model evaluation.

SARCTwitter. To test our models on comments with a structure more similar to the EmoNet ones, we select the benchmark dataset used by Majumder et al. [28] and created by Mishra et al. [7]. The dataset consists of 994 tweets from Twitter, manually annotated by seven readers with both sarcastic and sentiment information, i.e., each tweet has two labels, one for sentiment and one for sarcasm. Out of 994 tweets, 383 are labeled as positive (sentiment) and the remaining 611 are labeled as negative (sentiment). Additionally, out of these 994 tweets, 350 are labeled as sarcastic and the remaining 644 are labeled as non-sarcastic. The dataset contains also eye-movement of the readers that we ignored for our experiment as our focus is to detect sarcasm *solely* from the text content. We refer to this dataset as SARCTwitter.

5. Experiments

5.1. Implementation Details

To obtain a reliable and well-performing model, we studied a supervised learning approach on the three sarcasm datasets. We implement our models using the AllenNLP library [42] and HuggingFace Transformers library [36]. To perform our experiments we use the AWS Platform, EC2 instances (Ubuntu Deep Learning AMI) with one GPU on a PyTorch environment.

Each input sentence is passed through our pre-trained Base Uncased BERT. We then utilize the semantic content in the first special token `[CLS]` and feed it into a linear layer. We then apply softmax [43] to compute the class probability and output the label with the highest probability.

We iterate over each dataset with a mini-batch of size 16. We use AdaGrad optimizer [44] having gradient clipping threshold set to 5.0. We tune hyper-parameters on the validation set of each dataset. For every epoch we compute F1-score and Accuracy. The training is stopped (for both target task and intermediate tasks) once the average F1 on the validation set ceases to grow after some consecutive epochs (the patience is set to 5).

Table 6 shows the performance of BERT with intermediate task fine-tuning on the validation set for each task. The corresponding BERT models were transferred and further fine-tuned on the target task.

Table 6. Results on the Intermediate Tasks.

Intermediate Task	Avg_F1	Accuracy
EmoNet	49.11	60.65
EmoNetSent	84.64	84.69
IMDB	93.58	93.58

5.2. Experiments on Sarcasm V2 Corpus

Oraby et al. [11] performed supervised learning using SVM and, as the Sarcasm V2 dataset has a small size, they executed a 10-fold cross-validation on the data to obtain the state-of-the-art metrics shown in the results section. For our approach, we randomly divided the Gen Dataset into 90% training and 10% test set. Then, we split the temporary training set into 80% training and 20% validation set. We performed this procedure five times using a different random seed each time, obtaining five sets of data. Note that, as we can see from Table 7, all subsets were maintained balanced (i.e., with the same number of sarcastic and non-sarcastic data). We ran the same model over the five created splits and computed the mean values of the metrics that we obtained through the five executions.

Table 7. Sarcasm V2 dataset size.

Set	Sarcastic	Not Sarcastic	Total
Training	2348	2348	4696
Validation	586	586	1172
Test	326	326	652

5.3. Experiments on SARC

On the SARC dataset, as Khodak et al. [18] provided also a balanced test set for the training task, we only had to create our own validation set. We first removed some noise data from our training. Specifically, about 40 empty comments were found and deleted (which is equivalent to a really small percentage of the dataset–a negligible quantity that cannot affect our models' performance). We then divided our original training set into 80% training and 20% validation. Both collections have been shuffled and maintained balanced. We performed our evaluation with the same models used in the Sarcasm V2 experiments and compared our performance with previous works. Table 8 shows the size of each of the subsets used in our experiments.

Table 8. SARC Main Balanced dataset size.

Set	Sarcastic	Non-Sarcastic	Total
Original Training	505,390	505,390	1,010,780
Training	404,312	404,312	808,624
Validation	101,078	101,078	202,156
Test	125,804	125,804	251,608

5.4. Experiments on SARCTwitter

For the SARCTwitter dataset, we used an approach similar to the Sarcasm V2 Corpus. Unlike the above two datasets, the one by Mishra et al. [7] is not balanced, i.e., there are more non-sarcastic tweets than sarcastic ones. So, we decided to randomly split the dataset five times, keeping unchanged the ratio between the sarcastic and non-sarcastic tweets (as in the original set). Similarly to Sarcasm V2, we split the initial 994 tweets into 90% training and 10% test set. Then, we split again the obtained training set into 80% training and 20% validation, keeping always unchanged the ratio between the labels (see Table 9). We experimented with this dataset using all the baselines from the previous experiments.

Table 9. SARCTwitter dataset size.

Set	Sarcastic	Not Sarcastic	Total
Training	251	464	715
Validation	63	115	178
Test	35	63	98

6. Results

In this section, we discuss prior works for each dataset and present comparison results.

6.1. Results on Sarcasm V2 Corpus

6.1.1. Prior Works

State-of-the-art on this dataset is obtained by Oraby et al. [11]. The authors run their experiment using the following models:

- SVM-W2V: An SVM classifier with Google News Word2Vec (W2V) [45] embeddings as features to capture semantic generalizations.
- SVM-N-grams: An SVM classifier with N-grams features, including unigrams, bi-grams, and trigrams, sequences of punctuation and emoticons.

We show their state-of-the-art results in terms of F1-score computed on the Sarcastic label, on the last row of Table 10. To better underline the performance of BERT, we added also our own baseline models as terms of comparison, a simple BiLSTM and a CNN encoder, both fed with pre-trained contextualized ELMo embeddings trained on the 1 Billion Word Benchmark (https://www.statmt.org/lm-benchmark/, accessed on 19 March 2021) (approximately 800M tokens of news crawl data from WMT 2011 http://www.statmt.org/wmt11/translation-task.html, accessed on 19 March 2021).

6.1.2. Analysis and Discussion on Sarcasm V2

The results in Table 10 reveal that all our experiments outperform the existing state-of-the-art for the Sarcasm V2 Corpus. Our BiLSTM and CNN baselines, that obtain similar performance with each other, exceed the previous state-of-the-art by 2% and they are outperformed by our simple BERT model by 4%. These results prove the efficacy of neural models with word embeddings over feature engineering methods with SVMs. The transfer models, except for TransferEmoNet, reach similar results. The lower performance of TransferEmoNet can be explained by the scarce emotion distribution in IAC. In fact, the Sarcasm V2 comments are mainly responses to debates, in which emotions such as fear and anticipation are very rare.

Table 10. Results on the Sarcasm V2 dataset. Bold font shows best performance overall.

Model	F1-Score
BERT (no intermediate pre-training)	80.59
BERT + TransferEmoNet	78.56
BERT + TransferEmoNetSent	80.58
BERT + TransferIMDB	**80.85**
BiLSTM (ELMo)	76.03
CNN (ELMo)	76.46
SVM with N-Grams (Oraby et al. [11])	72.00
SVM with W2V (Oraby et al. [11]) (SOTA)	74.00

In this experiment, the model pre-trained on the IMDB dataset achieves state-of-the-art performance, outperforming the vanilla BERT model by 0.3%. The increase may be explained by the fact that the features of the Sarcasm V2 comments are more similar to the ones of movie reviews rather than tweets. That is, they are much longer in length than the

tweets' lengths and this difference in lengths brings additional challenges to the models. The expressions of sentiment/emotions in EmoNet, i.e., lexical cues, are more obvious in EmoNet compared with IMDB. Thus, the model struggles more on the IMDB dataset, and hence, is able to learn better and more robust parameters since these examples are more challenging to the model, and therefore, more beneficial for learning. This also explains the lack of improvement for the TransferEmoNetSent model. However, the outcomes of this experiment underline that there is correlation between sarcasm and sentiment. BERT is able to outperform previous approaches on this dataset and acts as a strong baseline. Using BERT with intermediate task transfer learning can push the performance further.

6.2. Results on SARC

6.2.1. Prior Works

We compared our best models with state-of-the-art networks and baselines examined by Hazarika et al. [27] on the Main Balanced version of SARC:

- Bag-of-words: A model that uses an SVM having a comment's word counts as features.
- CNN: A simple CNN that can only model the content of a comment.
- CNN-SVM: A model developed by Poria et al. [25] that exploits a CNN to model the content of the comments and other pre-trained CNNs to extract sentiment, emotion, and personality features from them. All these features are concatenated and passed to an SVM to perform classification.
- CUE-CNN: A method proposed by Amir et al. [13] that also models user embeddings combined with a CNN thus forming the CUE-CNN model.
- Bag-of-Bigrams: A previous state-of-the-art model for this dataset, by Khodak et al. [18], that uses the count of bigrams in a document as vector features.
- CASCADE (ContextuAl SarCAsm DEtector): A method proposed by Hazarika et al. [27] that uses user embeddings to model user personality and stylometric features, and combines them with a CNN to extract content features. We show the results from both versions, with and without personality features, in order to emphasize the efficacy of our model even in the absence of user personality feature.

6.2.2. Analysis and Discussion on SARC

Table 11 shows the results of our models and of the described prior works in terms of F1-score. The table has been divided into two sections: the first section contains all the experiments that have been run on the sentences themselves without the use of any additional information, while the second part contains the performance of models that exploit personality features from the authors of the comments. Since all our models, including our BiLSTM baseline, do not use author information, they appear in the first section of the table.

We can notice that in the first section of the table, all our models outperform all the other prior works by at least 10% confirming the efficacy of capturing semantics through the pre-trained language models for the sarcasm prediction task. In addition, our simplest model trained with BERT Base outperforms all the previous works, including the previous state-of-the-art CASCADE, that makes use of personality features. Similar to the Sarcasm V2 experiments, here the transfer-learning improves the performance of BERT base model only slightly.

Table 11. Results on the SARC dataset. Bold font shows best performance overall.

Models	F1-Score
No personality features	
BERT (no intermediate pre-training)	77.49
BERT + TransferEmoNet	77.22
BERT + TransferEmoNetSent	**77.53**
BERT + TransferIMDB	77.48
BiLSTM (ELMo)	76.27
Bag-of-words (Hazarika et al. [27])	64.00
CNN (Hazarika et al. [27])	66.00
CASCADE (Hazarika et al. [27]) (no personality features)	66.00
With personality features	
CNN-SVM (Poria et al. [25])	68.00
CUE-CNN (Amir et al. [13])	69.00
CASCADE (Hazarika et al. [27]) (with personality features) (SOTA)	77.00

This behavior can be explained by the fact that comments from discussion forums, such as Reddit, are quite different in terms of content, expressiveness, and topic from the other social platforms of our intermediate tasks. For example, SARC comments' lengths can vary from 3/4 words to hundreds, while the IMDB movie reviews are generally much longer, composed of multiple sentences, whereas EmoNet tweets usually consist of just one or two sentences. In addition, on EmoNet the sentiment pattern is more pronounced as people are more prone to describe their emotional state on Twitter. In SARC, probably also because of the topics covered (e.g., politics, videogames), the emotion pattern is more implicit and harder to detect. In the movie reviews, on the other hand, the sentiment is quite explicit but the length of the sentences may cause a loss of information for the classifier and the sarcastic content is almost nonexistent. However, the sentiment information from EmoNet slightly improved the efficacy of the simple BERT classification, making our TransferEmoNetSent model the new state-of-the-art performance on the SARC dataset.

These results support the pattern discovered on the Sarcasm V2 dataset, highlighting BERT as the best-performing model and underlining the importance of sentiment in sarcasm classification. This statement will be confirmed by our last experiment.

6.3. Results on SARCTwitter

6.3.1. Prior Works

We compared the BERT models with previous works provided by Mishra et al. [29]:

- CNN only text: A CNN-based framework that classifies tweets using only the information provided in their text.
- CNN gaze + text: An advanced framework developed by Mishra et al. [29] that adds cognitive features obtained from the eye-movement/gaze data of human readers to the previous CNN-based classifier in order to detect the sarcastic content of the tweets.
- GRU+MTL: The current state-of-the-art method by Majumder et al. [28] that uses multitask learning with sarcasm detection as the main task and sentiment classification as an auxiliary task. They used a GRU-based neural network with an attention mechanism.

6.3.2. Analysis and Discussion on SARCTwitter

From Table 12, we can see that all our models outperform the previous state-of-the-art by at least 5%. The table underlines how our models are able to accurately detect sarcasm and proves the strength of large pre-trained language models for this task. In particular,

even from this experiment, BERT is shown to be the most suitable model for sarcasm detection, outperforming the BiLSTM model by more than 1%.

Table 12. Results on SARCTwitter dataset. Bold font shows best performance overall.

Model	F1-Score
BERT (no intermediate pre-training)	96.34
BERT + TransferEmoNet	96.71
BERT + TransferEmoNetSent	**97.43**
BERT + TransferIMDB	95.96
BiLSTM (ELMo)	95.10
CNN only text (Mishra et al. [29])	85.63
CNN gaze + text (Mishra et al. [29])	86.97
GRU+MTL (Majumder et al. [28]) (SOTA)	90.67

Furthermore, unlike the previous experiments, where the addition of an intermediate task caused only slight improvements (no more than 0.3%) over the vanilla BERT model, here, the transfer learning models, especially on EmoNetSent, show improvement ($\sim$1% for EmoNetSent) over vanilla BERT. Indeed, the performance of vanilla BERT is close to that of BERT with intermediate task transfer learning on Sarcasm V2 and SARC, but we can see a larger improvement in performance between these two models on the SARCTwitter dataset (see Table 12). A potential reason for this is the size of these datasets. Both Sarcasm V2 and SARC are larger in size compared with SARCTwitter (e.g., SARC has 1M examples in the training set and the models are already well trained on this dataset and able to learn robust model parameters). The main purpose of intermediate task transfer learning is to use data-rich sources of relevant information (e.g., sentiment) when the dataset for the target task (i.e., sarcasm in our case) is small in size. Our results validate the fact that when the dataset size is small (e.g., as is the case with the SARCTwitter dataset) using BERT with intermediate task transfer learning achieves a substantially better performance compared with vanilla BERT. Interestingly, although the improvement of intermediate task transfer learning is very small on the SARC dataset, given the considerable size of SARC (i.e., 1M examples), in this SARC dataset even a small increase may be considered relevant.

Another potential reason for the increased performance of BERT with intermediate task transfer learning over vanilla BERT on SARCTwitter is that the EmoNet intermediate models are trained from the same social media domain and are rich in polarized emotions (positive/negative) that are useful for the detection of sarcasm. In fact, as this dataset structure is similar to the EmoNet one (i.e., short sentences from Twitter), both the emotion and sentiment information help improve the performance of the sarcasm classification task. On this dataset, our TransferEmoNetSent model reaches state-of-the-art performance, outperforming the previous state-of-the-art by almost 7% and boosting the simple BERT model's efficacy by more than 1%. In contrast, BERT + TransferIMDB performs worse than the vanilla BERT. We believe that this happens because of the domain (platform) mismatch (e.g., short text vs. longer text, more implicit vs. more explicit mentions of sarcasm or sentiment/polarized emotions).

These results confirm the pattern of the previous experiments, proving the correlation between sarcasm and sentiment, and also show that polarized emotional information can help the primary/target task with transfer from datasets where the emotional charge is more explicit, such as EmoNet which is annotated using distant supervision using lexical surface patterns [38].

7. Conclusions and Future Work

Sarcasm is a complex phenomenon which is often hard to understand, even for humans. In our work, we showed the effectiveness of using large pre-trained BERT language models to predict it accurately. We demonstrated how sarcastic statements themselves

can be recognized automatically with a good performance without even having to further use contextual information, such as users' historical comments or parent comments. We also explored a transfer learning framework to exploit the correlation between sarcasm and the sentiment or emotions conveyed in the text, and found that an intermediate task training on a correlated task can improve the effectiveness of the base BERT models, with sentiment having a higher impact than emotions on the performance, especially on sarcasm detection datasets that are small in size. We thus established new state-of-the-art results on three datasets for sarcasm detection. Specifically, the improvement in performance of BERT-based models (with and without intermediate task transfer learning) compared with previous works on sarcasm detection is significant and is as high as 11.53%. We found that the BERT models that use only the message content perform better than models that leverage additional information from a writer's history encoded as personality features in prior work. We found this result to be remarkable. Moreover, if the dataset size for the target task—sarcasm detection—is small then intermediate task transfer learning (with sentiment as the intermediate task) can improve the performance further.

We believe that our models can be used as strong baselines for new research on this task and we expect that enhancing the models with contextual data, such as user embeddings, in future work, new state-of-the-art performance can be reached. Integrating multiple intermediate tasks at the same time could potentially improve the performance further, although caution should be taken to avoid the loss of knowledge from the general domain while learning from the intermediate tasks. We make our code available to further research in this area.

Author Contributions: Both authors E.S. and C.C. contributed ideas and the overall conceptualization of the project. E.S. wrote the code/implementation of the project and provided an initial draft of the paper. Both authors E.S. and C.C. worked on the writing and polishing of the paper and addressed reviewers' comments. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Science Foundation (NSF) and Amazon Web Services under grant number NSF-IIS: BIGDATA #1741353. Any opinions, findings, and conclusions expressed here are those of the authors and do not necessarily reflect the views of NSF.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in our experiments are available online at the following links: the Sarcasm V2 Corpus at https://nlds.soe.ucsc.edu/sarcasm2 (accessed on 23 March 2021), SARC at http://nlp.cs.princeton.edu/SARC/ (accessed on 23 March 2021), and SARCTwitter at http://www.cfilt.iitb.ac.in/cognitive-nlp/ (accessed on 29 March 2021).

Acknowledgments: We thank our anonymous reviewers for their constructive comments and feedback, which helped improve our paper. This research is supported in part by the National Science Foundation and Amazon Web Services (for computing resources).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Maynard, D.; Greenwood, M. Who cares about Sarcastic Tweets? Investigating the Impact of Sarcasm on Sentiment Analysis. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, 26–31 May 2014; pp. 4238–4243.
2. Sykora, M.; Elayan, S.; Jackson, T.W. A qualitative analysis of sarcasm, irony and related #hashtags on Twitter. *Big Data Soc.* **2020**, *7*, doi:10.1177/2053951720972735.
3. Joshi, A.; Sharma, V.; Bhattacharyya, P. 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 (Short Papers), Beijing, China, 26–31 July 2015; Volume 2.
4. Wallace, B.C.; Choe, D.K.; Kertz, L.; Charniak, E. 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 (Short Papers), Baltimore, MD, USA, 22–27 June 2014; Volume 2, pp. 512–516. https://doi.org/10.3115/v1/P14-2084.

5. Oraby, S.; El-Sonbaty, Y.; Abou El-Nasr, M. Exploring the Effects of Word Roots for Arabic Sentiment Analysis. In Proceedings of the Sixth International Joint Conference on Natural Language Processing, Nagoya, Japan, 14–19 October 2013; pp. 471–479.
6. Riloff, E.; Qadir, A.; Surve, P.; De Silva, L.; Gilbert, N.; Huang, R. Sarcasm as Contrast between a Positive Sentiment and Negative Situation. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Seattle, WA, USA, 18–21 October 2013; pp. 704–714.
7. Mishra, A.; Kanojia, D.; Bhattacharyya, P. Predicting readers' sarcasm understandability by modeling gaze behavior. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–16 February 2016.
8. Khodak, M.; Risteski, A.; Fellbaum, C.; Arora, S. Automated WordNet Construction Using Word Embeddings. In Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and Their Applications, Valencia, Spain, 4 April 2017; pp. 12–23. https://doi.org/10.18653/v1/W17-1902.
9. Oprea, S.V.; Magdy, W. iSarcasm: A Dataset of Intended Sarcasm. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 1279–1289. Available online: https://aclanthology.org/2020.acl-main.118/ (accessed on 12 February 2021).
10. Chauhan, D.S.; Dhanush, S.R.; Ekbal, A.; Bhattacharyya, P. Sentiment and Emotion help Sarcasm? A Multi-task Learning Framework for Multi-Modal Sarcasm, Sentiment and Emotion Analysis. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 4351–4360. https://doi.org/10.18653/v1/2020.acl-main.401.
11. Oraby, S.; Harrison, V.; Reed, L.; Hernandez, E.; Riloff, E.; Walker, M. Creating and Characterizing a Diverse Corpus of Sarcasm in Dialogue. In Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Los Angeles, CA, USA, 13–15 September 2016; pp. 31–41.
12. Liebrecht, C.; Kunneman, F.; van den Bosch, A. The perfect solution for detecting sarcasm in tweets #not. In Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Atlanta, GA, USA, 14 June 2013; pp. 29–37.
13. Amir, S.; Wallace, B.C.; Lyu, H.; Silva, P.C.M.J. Modelling context with user embeddings for sarcasm detection in social media. *arXiv* **2016**, arXiv:1607.00976.
14. Ghosh, A.; Veale, T. Fracking sarcasm using neural network. In Proceddings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, San Diego, CA, USA, 16 June 2016.
15. Joshi, A.; Tripathi, V.; Patel, K.; Bhattacharyya, P.; Carman, M. Are word embedding-based features useful for sarcasm detection? *arXiv* **2016**, arXiv:1610.00883.
16. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
17. Pruksachatkun, Y.; Phang, J.; Liu, H.; Htut, P.M.; Zhang, X.; Pang, R.Y.; Vania, C.; Kann, K.; Bowman, S.R. Intermediate-Task Transfer Learning with Pretrained Models for Natural Language Understanding: When and Why Does It Work? *arXiv* **2020**, arXiv:2005.00628.
18. Khodak, M.; Saunshi, N.; Vodrahalli, K. A Large Self-Annotated Corpus for Sarcasm. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018.
19. Kreuz, R.J.; Caucci, G.M. Lexical influences on the perception of sarcasm. In Proceedings of the Workshop on computational approaches to Figurative Language, Rochester, NY, USA, 26 April 2007; pp. 1–4.
20. Carvalho, P.; Sarmento, L.; Silva, M.J.; De Oliveira, E. Clues for detecting irony in user-generated contents: oh...!! it's so easy;-). In Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion, Hong Kong, China, 6 November 2009; pp. 53–56.
21. Tsur, O.; Davidov, D.; Rappoport, A. ICWSM—A great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, Washington, DC, USA, 23–26 May 2010.
22. Davidov, D.; Tsur, O.; Rappoport, A. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In Proceedings of the Fourteenth Conference on Computational Natural Language Learning, Uppsala, Sweden, 15–16 July 2010; pp. 107–116.
23. Rajadesingan, A.; Zafarani, R.; Liu, H. Sarcasm detection on twitter: A behavioral modeling approach. In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, Shanghai, China, 2–6 February 2015; pp. 97–106.
24. Bamman, D.; Smith, N.A. Contextualized sarcasm detection on twitter. In Proceedings of the Ninth International AAAI Conference on Web and Social Media, Oxford, UK, 26–29 May 2015.
25. Poria, S.; Cambria, E.; Hazarika, D.; Vij, P. A deeper look into sarcastic tweets using deep convolutional neural networks. *arXiv* **2016**, arXiv:1610.08815.
26. Zhang, M.; Zhang, Y.; Fu, G. Tweet sarcasm detection using deep neural network. In Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics, Osaka, Japan, 11–16 2016; pp. 2449–2460.
27. Hazarika, D.; Poria, S.; Gorantla, S.; Cambria, E.; Zimmermann, R.; Mihalcea, R. CASCADE: Contextual Sarcasm Detection in Online Discussion Forums. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 20–26 August 2018; pp. 1837–1848.
28. Majumder, N.; Poria, S.; Peng, H.; Chhaya, N.; Cambria, E.; Gelbukh, A.F. Sentiment and Sarcasm Classification with Multitask Learning. *arXiv* **2019**, arXiv:1901.08014.

29. Mishra, A.; Dey, K.; Bhattacharyya, P. Learning cognitive features from gaze data for sentiment and sarcasm classification using convolutional neural network. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July 2017–4 August 2017; pp. 377–387.
30. Plepi, J.; Flek, L. Perceived and Intended Sarcasm Detection with Graph Attention Networks. In *Findings of the Association for Computational Linguistics: EMNLP 2021*; Association for Computational Linguistics: Punta Cana, Dominican Republic, 2021; pp. 4746–4753. https://doi.org/10.18653/v1/2021.findings-emnlp.408.
31. Cai, Y.; Cai, H.; Wan, X. Multi-Modal Sarcasm Detection in Twitter with Hierarchical Fusion Model. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July 2019–2 August 2019; pp. 2506–2515. https://doi.org/10.18653/v1/P19-1239.
32. Li, L.; Levi, O.; Hosseini, P.; Broniatowski, D. A Multi-Modal Method for Satire Detection using Textual and Visual Cues. In Proceedings of the 3rd NLP4IF Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda, Barcelona, Spain, 20 December 2020; pp. 33–38.
33. Wang, X.; Sun, X.; Yang, T.; Wang, H. Building a Bridge: A Method for Image-Text Sarcasm Detection Without Pretraining on Image-Text Data. In Proceedings of the First International Workshop on Natural Language Processing Beyond Text, Online, 20 November 2020; pp. 19–29. https://doi.org/10.18653/v1/2020.nlpbt-1.3.
34. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. https://doi.org/10.1109/CVPR.2016.90.
35. Lu, J.; Batra, D.; Parikh, D.; Lee, S. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
36. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv* **2019**, arXiv:1910.03771.
37. Phang, J.; Févry, T.; Bowman, S.R. Sentence Encoders on STILTs: Supplementary Training on Intermediate Labeled-data Tasks. *arXiv* **2018**, arXiv:1811.01088.
38. Abdul-Mageed, M.; Ungar, L. EmoNet: Fine-Grained Emotion Detection with Gated Recurrent Neural Networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Long Papers), Vancouver, BC, Canada, 30 July 2017–4 August 2017; Volume 1, pp. 718–728. https://doi.org/10.18653/v1/P17-1067.
39. Maas, A.L.; Daly, R.E.; Pham, P.T.; Huang, D.; Ng, A.Y.; Potts, C. Learning Word Vectors for Sentiment Analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; pp. 142–150.
40. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
41. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1746–1751. https://doi.org/10.3115/v1/D14-1181.
42. Gardner, M.; Grus, J.; Neumann, M.; Tafjord, O.; Dasigi, P.; Liu, N.; Peters, M.; Schmitz, M.; Zettlemoyer, L. Allennlp: A deep semantic natural language processing platform. *arXiv* **2018**, arXiv:1803.07640
43. Bridle, J.S. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing*; Springer: Berlin/Heidelberg, Germany, 1990; pp. 227–236.
44. Duchi, J.; Hazan, E.; Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* **2011**, *12*, 2121–2159.
45. Mikolov, T.; Chen, K.; Corrado, G.S.; Dean, J. Efficient Estimation of Word Representations in Vector Space. In Proceedings of the 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, AZ, USA, 2–4May 2013; Workshop Track Proceedings.