Does Commonsense help in detecting Sarcasm?

Somnath Basu Roy Chowdhury **Snigdha Chaturvedi**
{somnath, snigdha}@cs.unc.edu
UNC Chapel Hill

Abstract

Sarcasm detection is important for several NLP tasks such as sentiment identification in product reviews, user feedback, and online forums. It is a challenging task requiring a deep understanding of language, context, and world knowledge. In this paper, we investigate whether incorporating commonsense knowledge helps in sarcasm detection. For this, we incorporate commonsense knowledge into the prediction process using a graph convolution network with pre-trained language model embeddings as input. Our experiments with three sarcasm detection datasets indicate that the approach does not outperform the baseline model. We perform an exhaustive set of experiments to analyze where commonsense support adds value and where it hurts classification. Our implementation is publicly available at: https://github.com/brcsomnath/commonsense-sarcasm.

1 Introduction & Related Work

The topic of sarcasm has received attention in various research fields like linguistics (Utsumi, 2000), psychology (Gibbs, 1986; Kreuz and Glucksberg, 1989) and the cognitive sciences (Gibbs Jr et al., 2007). Identifying sarcasm is essential to understanding the opinion and intent of a user in downstream tasks like opinion mining, sentiment classification, etc. Initial approaches for this task (Kreuz and Glucksberg, 1989) mostly relied on handcrafted features to capture the lexical and contextual information. On similar lines, the efficacy of special characters, emojis and n-gram features in the discrimination task have also been studied (Carvalho et al., 2009; Lukin and Walker, 2013).

In recent years, this task has gained traction in the machine learning and computational linguistic community (Davidov et al., 2010; González-Ibáñez et al., 2011; Riloff et al., 2013; Maynard and Greenwood, 2014; Wallace et al., 2014; Ghosh et al., 2015; Joshi et al., 2015; Muresan et al., 2016; Amir et al., 2016; Mishra et al., 2016; Ghosh and Veale, 2017; Chakrabarty et al., 2020). Several approaches have studied the role of context in this sarcasm detection task (Ghosh et al., 2020). However, none of the previous works have explored the idea of incorporating commonsense knowledge in sarcasm detection. Common sense has been used in several natural-language based tasks like controllable story generation (Zhang et al., 2020; Brahman and Chaturvedi, 2020), sentence classification (Chen et al., 2019), question answering (Dzendzik et al., 2020), natural language inference (K M et al., 2018; Wang et al., 2019) and other related tasks but not for sarcasm detection. We hypothesize that commonsense knowledge, capturing general beliefs and world knowledge, can prove instrumental in understanding sarcasm. For example in Figure 1, for the event "I loved the movie so much that I left during the interval" (an example of sarcasm with polarity contrast), we show how commonsense is able to capture the contrast between the intentions of the subject before and during the event. Incorporating such commonsense knowledge ideally should make it easier for the learning model to detect sarcasm where it is not apparent from the input.

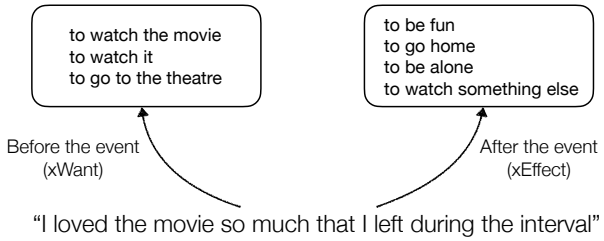

Figure 1: COMET output for the sentence "I loved the movie so much that I left during the interval". The commonsense sequences capture the contrast between intent and action of the subject.

With this motivation, we study the utility of common sense information for sarcasm detection. For

Proceedings of the Second Workshop on Insights from Negative Results in NLP, pages 9–15

ISBN 978-1-954085-93-0

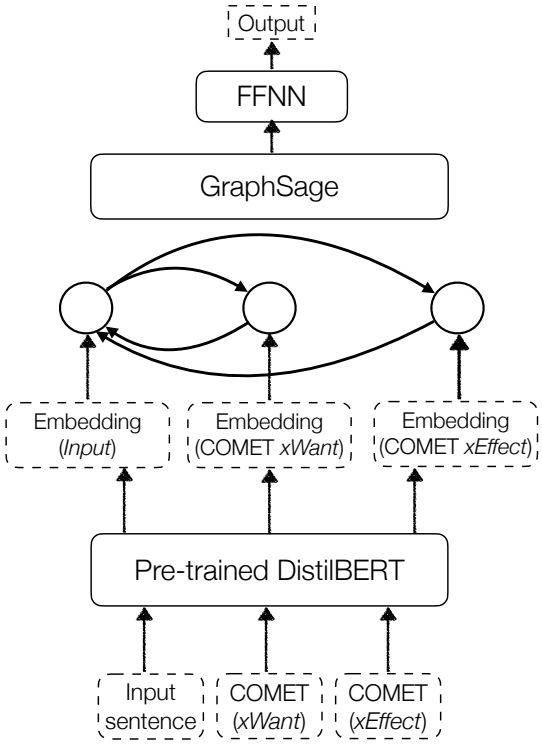

Figure 2: Proposed model architecture. Representations of the input sentence along with two COMET sequences are retrieved from pre-trained DistilBERT that are used to initialize a GCN. Post training, node representations of the graph is passed through a fully-connected neural network to generate the output.

this, we leverage COMET (Bosselut et al., 2019) to extract the relevant social commonsense information for a sentence. Given an event, COMET provides likely scenarios relating to various attributes like intent of the subject, effect on the object etc.

We use a GCN (Kipf and Welling, 2017) based model for infusing commonsense knowledge in the sarcasm detection task. Our experiments reveal that the commonsense augmented model performs at par with the baseline model. We perform an array of analysis experiments to identify where the commonsense infused model outperforms the baseline and where it fails.

2 Model

We use a graph convolution-based model to enable incorporation of COMET sequences given an input sentence. The sentence representations are retrieved from the pre-trained encoder of our baseline model. Our **baseline model** consists of a Transformer (Vaswani et al., 2017) based DistilBERT (Sanh et al., 2019) encoder followed by a feed-forward neural network (FFNN). DistilBERT is a light-weight encoder, which enables faster training, while achieving similar performance as other Transformer based encoders.

DATASET	TRAINING	TEST
SemEval Irony	3833	958
News Headline	27691	27691
FigLang 2020 Reddit	3520	880

Table 1: Number of training samples in train/test split for each dataset.

The model is shown in Figure 2. For every input instance, a graph is formed with edges between the input sentence and COMET sequences. No edges are present between individual COMET sequences. Sentence embeddings retrieved from the baseline DistilBERT form the initial graph embeddings. The intuition behind leveraging a graph-based architecture was to enable information flow between the representations of the input sentence and COMET sequences, thereby reducing the domain discrepancy between them.

The graph is then fed into a GraphSage (Hamilton et al., 2017) network which produces the node embedding vector $V \in \mathbb{R}^{(M+1)\times N}$, where M is the number of COMET sequences and N is the output dimension of the GCN. The node embedding vector V is then forwarded to a fully connected neural network layer to produce the final output. In section 4, we experiment with different edge configurations and observe how each edge configuration affects the downstream performance.

We experimented with another model that incorporated COMET sequences with an attention-mechanism. In that model, the representation of the input sentence was concatenated with an aggregate representation of the COMET sequences, formed in an attentive fashion. Its performance was not better than the GCN-based model, so we do not describe it here.

3 Experimental Setup

We evaluate the models on three datasets (a) **Irony detection SemEval task**: Van Hee et al. (2018) conducted a SemEval task for irony detection considering an utterance in isolation. They also released a secondary task where the sarcastic samples were classified into *three* broad categories: verbal irony with polarity contrast, situational irony, and others. (b) **News Headlines dataset** (Misra and

APPROACH	News Headlines	SemEval Irony	FigLang 2020 (Reddit)
Baseline	96.13%	69.09%	67.95%
GCN (*bidirectional* edges)	96.16%	67.88%	67.50%
GCN (input → COMET edges)	96.14%	68.66%	67.35%
GCN (COMET → input edges)	96.18%	68.40%	67.54%

Table 2: Accuracy of the baseline DistilBERT and GCN model (in various edge configurations). We do not observe any significant change in sarcasm detection performance with the incorporation of commonsense sequences.

Edge configuration	Performance
GCN (*bidirectional*)	67.27%
GCN (COMET → input)	55.00%
GCN (input → COMET)	67.36%

Table 3: Performance of the proposed model for different edge configurations. We observe a sharp performance drop in (COMET → input) configuration.

Dataset	Overlap
News Headline	99.5%
SemEval Irony	91.6%
FigLang 2020 (Reddit)	92.7%

Table 4: Test set overlap where the output label from the GCN and DistilBERT model is the same.

Arora, 2019): contains sarcastic news headlines from *TheOnion* and non-sarcastic ones from *HuffPost*. (c) **FigLang 2020 Sarcasm detection task**: We experiment on the Reddit dataset of the shared task introduced by Ghosh et al. (2020). The statistics of the datasets are specified in Table 1.

All the aforementioned datasets are balanced. We report our results by randomly splitting into training and test set, and averaging the accuracy over 5 iterations. In our experiments, we incorporate a subset of COMET predicates (*xWant* and *xEffect*) related to the subject in a sentence.

4 Results

We report the classification accuracy of the models for all datasets in Table 2. The baseline denotes the DistilBERT performance. We see a high performance in the *News headline* dataset where the sentences are self-contained and language is not noisy. We see relatively lower performance of the baseline in FigLang 2020 Reddit, where we ignored the available context. Performance in SemEval dataset is low due to noisy tweets.

We conduct an ablation study with three configurations of the graph edges (a) *bidirectional* edges (b) edges from *input* → *COMET* sequences and (c) edges from *COMET sequences* → *input*. The results of the GCN-based model in different settings are shown in Table 2. The performance of the GCN model is at par with the baseline and varying edge configurations doesn't have any effect on the downstream performance. COMET produces sequences for an input event in the following format.
Input: "they should have put $125 million termination payout in each of their contract"
xWant : *to save money*
We wish to have more complete COMET sentences like: *they wanted to save money*. In order to improve the setup, we replace COMET sequences with complete sentences of the form: `[subject] [MASK] [raw COMET sequence]`. We replace the `[subject]` placeholder with the SUBJECT POS tag in the input sentence. We leverage a pre-trained BERT model to predict the unknown `[MASK]` word. All reported results use this setup.

We examine whether the COMET representations leverage information from the input in the GCN setup by **removing the input sentence representation** before the FFNN module (shown in Figure 2) and experimenting with different edge-configurations. In Table 3, we observe a significant performance dip with COMET→input setup. This illustrates that the information flowing from input sentence to COMET sequences is more relevant.

We also measure the share of instances in the test set having the same predicted label from the baseline and the model. We observe a significant overlap (>90%) between the predictions of the baseline and the proposed model across all datasets in Table 4, illustrating that the model isn't able to tackle new instances.

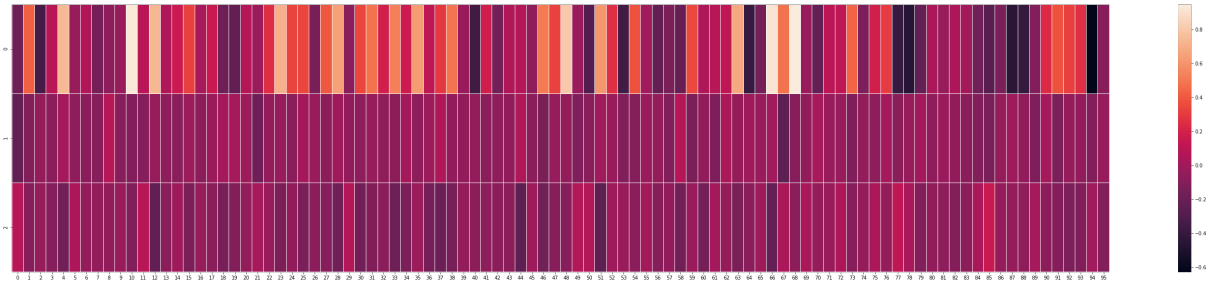

Figure 3: Visualization of gradient-based saliency tests. Darker shade denotes lower absolute values. The first row shows the features corresponding to the input sentence, and the other two rows are features from COMET sequences *xWant* and *xEffect*. We observe that features from input sentence (first row) receive high saliency values.

Occluded Element	Δ
Input sentence	27.99%
COMET sequences	1.38%

Table 5: Confidence change when different segments of the input are occluded. Δ denotes the change in confidence when different parts of the input is occluded.

DATASET	$\mathcal{C}^{NS}_{\text{GCN}}$
News Headline	83.8%
SemEval Irony	64.6%
FigLang 2020 Reddit	96.8%

Table 6: $\mathcal{C}^{NS}_{\text{GCN}}$ statistic across different datasets.

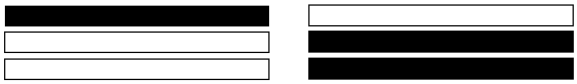

Figure 4: Occlusion setup. First setup shows that the input sentence representation (first row) is occluded. Second setup commonsense sequence representations are occluded (second and third rows).

5 Saliency Test

We perform saliency tests to investigate whether the model is reliant on commonsense sequences while taking decisions.

(a) **Gradient-based saliency** (Bastings and Filippova, 2020) measure for a feature x_i given an output class c is computed as $\nabla_{x_i}\mathcal{L}(y, f(x)) \cdot x_i$, where $\mathcal{L}(\cdot,\cdot)$ is the loss function. The saliency map is shown in Figure 3. The saliency map vector has a dimension of 3×768, where the first row showcases the saliency values of the features corresponding to the input sentence while the remaining two rows correspond to the saliency of COMET sequences. For better visualization, all values are normalized between 0-1 and average pooling is performed on adjacent blocks of 8 to form a vector of dimension 3×192. From Figure 3, it is evident that the model learns to identify important input features but assigns similar saliency values to all COMET features.

(b) **Occlusion-based saliency** test involves occluding a part of the input and observing the change in the output probability vector. We occlude the input representation and COMET representations respectively as shown in Figure 4. The occlusion metric (Bastings and Filippova, 2020) is defined as $\mathrm{E}_{x\sim\mathcal{D}}[|f_c(\mathbf{x}) - f_c(\mathbf{x}|x_i = 0)|]$. Table 5 reports the results of this test. We observe that occluding the input sentence leads to a significant change in the output confidence while occluding the COMET sequences has little impact.

These tests demonstrate that the model is more reliant on the input sentence and less on the COMET sequences for making the prediction.

6 Efficacy of Commonsense

In this section, we try to uncover why COMET sequences don't help in the sarcasm detection task. In order to identify instances where commonsense incorporation hurts the performance, we focus on samples where the model's prediction is wrong but the baseline is correct. Among these samples, we measure how many were non-sarcastic by defining a new measure **non-sarcastic class coverage**,

$$\mathcal{C}^{NS}_{\text{GCN}} = \frac{|\{x|x \in \mathcal{S}^{B}_{\text{GCN}}, l(x) = \mathcal{NS}\}|}{|\mathcal{S}^{B}_{\text{GCN}}|}$$

where $\mathcal{S}^{B}_{\text{GCN}}$ is the set of samples in which the model predicted incorrectly while the baseline was correct, $l(\cdot)$ is the oracle function which returns the true label of an input instance x, and $\mathcal{NS}$ denotes the non-sarcastic class label. Results in Ta-

Ground Truth	Input Sentence	COMET Support *(xWant and xEffect)*	Explanations
Non-sarcastic	*@usertag* i wonder if they have that in an audio book	• He gets to learns something new • He wants to be entertained	COMET sequences don't add value for classifying the non-sarcastic sample.
Sarcastic	Going to watch a movie about murder. merry christmas ;)	• The person wants to have fun • The person gets tired as a result	COMET sequences fail to explain the satire.
Sarcastic	final at 7am, I'm ready	• The person wants to go to bed • The person has to go to work	COMET captures the contrast between the intentions and results.
Sarcastic	As a girl my reason not to put on makeup is I'm satisfied with my face	• She wanted to look pretty • As a result she got complements	COMET doesn't provide relevant commonsense for capturing polarity contrast.

Table 7: Example input instances along with their ground truth label and corresponding commonsense sentences retrieved from COMET. We analyze the utility of COMET sequences described as explanations.

ble 6 demonstrate a high value of $\mathcal{C}^{NS}_{\text{GCN}}$ across all datasets, this indicates that the large fraction of the instances where the model was incorrect but the baseline was correct were non-sarcastic. After surveying non-sarcastic instances we infer that commonsense knowledge fails to explain non-sarcastic samples and is present as irrelevant context hurting downstream performance (Petroni et al., 2020).

There are cases where the prediction failed either due to noisy input (prevalent in the Twitter based SemEval dataset) or subtle play of words which COMET sequences fail to explain.

In order to investigate the utility of commonsense for specific type of sarcasm, we form a subset of the SemEval Irony dataset with samples only from irony with polarity contrast and non-sarcastic class by leveraging labels from the secondary SemEval task (mentioned in Section 3). $\mathcal{C}^{NS}_{\text{GCN}}$ for the new dataset is 57.1%, a significant reduction from the 64.6% in SemEval dataset in Table 6. We infer that commonsense is only useful in detecting sarcasm with polarity contrast but struggles with other types of sarcasm.

7 Qualitative Analysis

In this section, we analyze a few examples shown in Table 7 and observe whether the COMET sequences are helpful in detecting sarcasm. We have anonymized any twitter handle with "*@usertag*" to prevent any leak of private information.

- In the first example of Table 7, the input sentence is non-sarcastic. Retrieved commonsense sequences don't capture any information that may help in prediction.
- In several instances, a sentence is sarcastic due to a subtle play of words or use of language. The commonsense based model struggles in such scenarios as COMET sequences cannot explain such events as shown in the second instance of Table 7.
- In the third example of Table 7, we show that COMET sequences are able to perfectly capture the contrast between the intention and effect on the person.
- In rare cases like the fourth instance of Table 7, which is an example of irony with polarity contrast. It is still difficult for the commonsense model to explain the satire.

8 Conclusion and Future work

In this paper, we proposed the idea of integrating commonsense knowledge in the task of sarcasm detection. We observe that COMET infused model performs at par with the baseline. Through saliency tests, we observe that the model is less reliant on the commonsense representations in many cases. From our analysis, we infer that commonsense is most effective in identifying sarcasm with polarity contrast but fails to explain non-sarcastic samples or other types of sarcasm effectively, which hurts the overall performance. In the future, we will explore the utility of other forms of external knowledge such as factual world knowledge for sarcasm detection. We will also try to leverage commonsense to *explain* why a certain remark is sarcastic.

References

Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 167–177, Berlin, Germany. Association for Computational Linguistics.

Jasmijn Bastings and Katja Filippova. 2020. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 149–155, Online. Association for Computational Linguistics.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.

Faeze Brahman and Snigdha Chaturvedi. 2020. Modeling protagonist emotions for emotion-aware storytelling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5277–5294.

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

Tuhin Chakrabarty, Debanjan Ghosh, Smaranda Muresan, and Nanyun Peng. 2020. r^3: Reverse, retrieve, and rank for sarcasm generation with commonsense knowledge. *arXiv preprint arXiv:2004.13248*.

Jindong Chen, Yizhou Hu, Jingping Liu, Yanghua Xiao, and Haiyun Jiang. 2019. Deep short text classification with knowledge powered attention. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6252–6259. AAAI Press.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcasm in Twitter and Amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 107–116, Uppsala, Sweden. Association for Computational Linguistics.

Daria Dzendzik, Carl Vogel, and Jennifer Foster. 2020. Q. can knowledge graphs be used to answer boolean questions? a. it's complicated! Association for Computational Linguistics (ACL).

Aniruddha Ghosh and Tony Veale. 2017. Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 482–491, Copenhagen, Denmark. Association for Computational Linguistics.

Debanjan Ghosh, Weiwei Guo, and Smaranda Muresan. 2015. Sarcastic or not: Word embeddings to predict the literal or sarcastic meaning of words. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1003–1012, Lisbon, Portugal. Association for Computational Linguistics.

Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. 2020. A report on the 2020 sarcasm detection shared task. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 1–11, Online. Association for Computational Linguistics.

Raymond W Gibbs. 1986. On the psycholinguistics of sarcasm. *Journal of experimental psychology: general*, 115(1):3.

Raymond W Gibbs Jr, Raymond W Gibbs, and Herbert L Colston. 2007. *Irony in language and thought: A cognitive science reader*. Psychology Press.

Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in Twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586, Portland, Oregon, USA. Association for Computational Linguistics.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034.

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

Annervaz K M, Somnath Basu Roy Chowdhury, and Ambedkar Dukkipati. 2018. Learning beyond datasets: Knowledge graph augmented neural networks for natural language processing. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 313–322, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Roger J Kreuz and Sam Glucksberg. 1989. How to be sarcastic: The echoic reminder theory of verbal irony. *Journal of experimental psychology: General*, 118(4):374.

Stephanie Lukin and Marilyn Walker. 2013. Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. In *Proceedings of the Workshop on Language Analysis in Social Media*, pages 30–40, Atlanta, Georgia. Association for Computational Linguistics.

Diana Maynard and Mark Greenwood. 2014. 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)*, pages 4238–4243, Reykjavik, Iceland. European Language Resources Association (ELRA).

Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. 2016. Harnessing cognitive features for sarcasm detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1095–1104, Berlin, Germany. Association for Computational Linguistics.

Rishabh Misra and Prahal Arora. 2019. Sarcasm detection using hybrid neural network. *arXiv preprint arXiv:1908.07414*.

Smaranda Muresan, Roberto Gonzalez-Ibanez, Debanjan Ghosh, and Nina Wacholder. 2016. Identification of nonliteral language in social media: A case study on sarcasm. *Journal of the Association for Information Science and Technology*, 67(11):2725–2737.

Fabio Petroni, Patrick Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions. *arXiv preprint arXiv:2005.04611*.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714, Seattle, Washington, USA. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Akira Utsumi. 2000. Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony. *Journal of Pragmatics*, 32(12):1777–1806.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. SemEval-2018 task 3: Irony detection in English tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Byron C. Wallace, Do Kook Choe, Laura Kertz, and Eugene Charniak. 2014. 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, Baltimore, Maryland. Association for Computational Linguistics.

Xiaoyan Wang, Pavan Kapanipathi, Ryan Musa, Mo Yu, Kartik Talamadupula, Ibrahim Abdelaziz, Maria Chang, Achille Fokoue, Bassem Makni, Nicholas Mattei, et al. 2019. Improving natural language inference using external knowledge in the science questions domain. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7208–7215.

Mingda Zhang, Keren Ye, Rebecca Hwa, and Adriana Kovashka. 2020. Story completion with explicit modeling of commonsense knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 376–377.