

Unmasking Sarcasm - Enhancing Sentiment Analysis in E-Commerce Reviews and Questions

Hugo Bouy
Illinois Tech

hbouy@hawk.iit.edu

Rémi Kalbe
Illinois Tech

rkalbe@hawk.iit.edu

Mathias Roumane
Illinois Tech

mroumane@hawk.iit.edu

1. Problem description

In an increasingly e-commerce driven world, the analysis of review, comments and questions written by consumers online has become a field of interests. Understanding opinions and emotions expressed in product feedback/related questions is essential to enhance the user's experience. One aspect that might be overlooked in this area is sarcastic and humor detection that can lead to a misinterpretation of these texts.

Humor remains a complex human phenomenon that is far from having a clear definition. While humor and sarcasm mechanisms are integral to human interaction, its subjective nature makes it a challenging target for computational analysis. Recent work has been able to open up this area using deep learning and natural language processing advances. With this project, we will attempt to improve e-commerce review processing using deep learning models for humor disambiguation.

2. Brief Survey of Previous Work

Several studies have been conducted over the past years in the ambition of detecting humor and sarcasm:

Jain et al. [6] delved into the complexities of identifying sarcasm in Amazon reviews. Recognizing sarcasm is crucial for accurate sentiment analysis, especially since sarcastic comments can be misinterpreted by traditional opinion mining methods. They utilized the "Sarcasm Corpus" containing labeled ironic and regular Amazon reviews, extracting features like sentiment scores, punctuation patterns, and contextual elements that consider the contrast between review sentiment and product rating. Their experiments designated the Support Vector Machine (SVM) classifier as the most accurate, emphasizing the role of context in sarcasm detection.

Building upon the idea of sarcasm detection, Poria et al. [9] introduced a method using deep convolutional neural networks (CNNs). They critiqued traditional methods that treat sarcasm detection as mere text categorization, arguing that such approaches often miss the deeper understanding

of language nuances required for sarcasm. Their method integrates sentiment, emotion, and personality features extracted from pre-trained CNNs. By leveraging Twitter data, they contrasted sarcastic sentences with the ground-truth polarity of events. Their experiments with word embeddings from word2vec and a combined CNN-SVM approach demonstrated superior performance on benchmark datasets.

Yaghoobian et al. [11] further discussed the challenges of sarcasm detection in sentiment analysis. They categorized detection methods into content-based, which focus on lexical indicators, and context-based, which emphasize background knowledge. Their study highlighted the CASCADE model, which uses user embeddings to capture user-specific features, as an example of leveraging context for sarcasm detection.

Shifting the focus to humor detection, Ziser et al. [12] identified product bias in Product Question Answering (PQA) systems, where certain products attract more humorous questions. They proposed a deep-learning framework to detect humor in PQA, focusing on incongruity and subjectivity.

Annamoradnejad and Zoghi [1] proposed the ColBERT model for humor detection. This model leverages BERT embeddings for sentence representation and has achieved state-of-the-art results on various datasets.

Lastly, Gupta et al. [5] explored the potential of Large Language Models (LLMs) in humor detection. Their research emphasized the capability of LLMs to capture the intricacies associated with humor and offense detection.

3. Datasets

3.1. Combined dataset for sarcasm detection

To train our future models, we built a collection of datasets containing several sarcastic and non-sarcastic text. The main datasets used in the scope of this project are listed below.

- **Headlines dataset [7]:** Contains a list of 28,619 headlines collected from two news websites. On one hand,

TheOnions aims to produce sarcastic versions of real news events. On the other hand, real and non-sarcastic news headlines are collected from HuffPost. This dataset has the advantage of having no spelling mistakes and informal usage since it is written by dedicated professionals in a formal manner.

- **MUStARD++ dataset [2]:** Mustard++ is a multi-modal sarcasm dataset that has been annotated with 9 emotions. It was compiled from popular TV shows such as Friends, The Golden Girls or The Big Bang Theory. We will be using this dataset mostly to detect sarcasm but if we have time we may use the annotation to classify the emotion associated with the sarcastic sentence considered.
- **Sarcasm Corpus V2 dataset [8]:** This dataset contains both sarcastic and non-sarcastic utterances. They are additionally classified in three different types: generic (6,520 samples), hyperbole (1.164 samples) and rhetorical (1,702 samples).
- **Sarcasm Amazon reviews dataset [4]:** Contains a large number of both regular and ironic Amazon reviews. Each review is also associated with information about the product for which the review was written, the number of stars assigned by the author, etc. This dataset will be the most useful in the second part of our project when we link the sarcastic review with the information regarding the product.

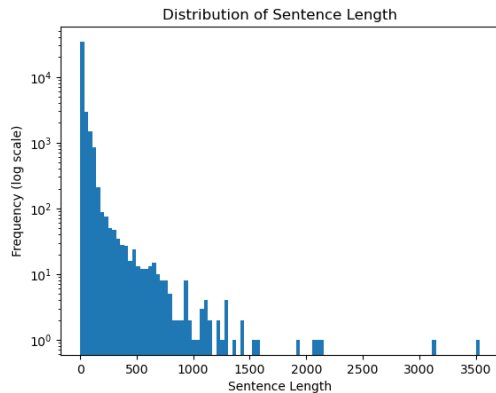


Figure 1. Frequency of sentence length in the final combined dataset

As an initial data processing step, we first concatenated those datasets together in order to use them to implement an initial sarcasm detection model. The final combined dataset then has the full sentence or review and a sarcastic indicator (0 or 1). The final combined dataset contains 40,461 samples from different sources and types. We then have a quite

complex dataset with sentences of very variable length as displayed in figure 1.

3.2. Amazon reviews dataset

In order to explore more in depth our problematic around e-commerce, the sarcasm Amazon reviews dataset remains the most useful one in the scope of our project. As described previously, it contains reviews of products on Amazon that are labelled either as sarcastic or regular. Additionally, this dataset reports other information associated to the review such as the product associated with it, the number of stars assigned by the author, etc. One of the sampled sarcastic reviews extracted from this dataset is given in table 1 as an example.

Feature	Example
STARS	3.0
TITLE	Great Product, Poor Packaging
DATE	May 14, 2009
AUTHOR	Patrick J. McGovern "Procrastinating Evil Scientist"
PRODUCT	Uranium Ore
REVIEW	I purchased this product 4.47 Billion Years ago and when I opened it today, it was half empty.

Table 1. Example of an ironic review sample and the associated information

The additional information offered by this dataset will allow us to take additional parameters into account rather than basing our prediction solely on the review. We consider that adding these features may help the model to better identify the underlying patterns of sarcasm.

4. Models

4.1. Simple Sarcasm detection model

As introduced above, we build a heterogeneous dataset with sarcasm examples from very different sources and context. Our goal here is to experiment building a model that could learn the inner structure of sarcasm from a large amount of data.

Before doing so, and as a starting point for this project, we first aimed at creating a simple model capable of detecting if a sentence contains some form of sarcasm using the Amazon review dataset only. At this stage, the model only takes into account the sentence and no additional factors. We will later add additional relevant factors in detecting sarcastic or humorous patterns such as information on the product considered.

Jain et al. previously conducted an experiment on this dataset [6]. Their idea was to build a feature vector for each Amazon review using various features such as the positive/negative sentiment score, punctuation, part of speech

and bigram analysis. Their results shown very good accuracy with Naïve Bayes (77.50%), a multi-layer perceptron classifier (81%), and SVM (81.5%), with similar precision, recall and f1. To verify the necessity of this feature selection, we created 2 models, one SVM and one LSTM and train them on plain data, without feature selection. SVM used count vectorization and the LSTM used word embedding. Our results showed that SVM performed well with 82% accuracy on the Amazon review dataset. However, digging into the results revealed that the model was more likely to misclassify a sarcasm text with both precision and recall being much under the ones for Regular text. One explanation for this difference may be the small size of the dataset which contains only 1254 texts, with only 437 being sarcastic.

	Precision	Recall	F1-score
Regular	0.84	0.90	0.87
Sarcasm	0.72	0.61	0.66

Table 2. Precision, Recall and F1-score for SVM on the Amazon review dataset with only count vectorization.

The LSTM model on its side less performed with a test accuracy of 67%. A Similar behavior can be observed regarding the sarcastic text for which the model fails to understand the sarcasm pattern in all the cases.

	Precision	Recall	F1-score
Regular	0.81	0.71	0.76
Sarcasm	0.43	0.57	0.49

Table 3. Precision, Recall and F1-score for LSTM on the Amazon review dataset with only count vectorization.

These results confirmed the necessity of feature selection before training the model.

Before applying feature selection we experimented with the state-of-the art model BERT over our entire dataset to observe if the wilder and more diverse example set of sarcasm could be a strong benefit.

4.2. BERT-based Sarcasm Detection Model on Combined Dataset

4.2.1 Introduction

Bidirectional Encoder Representations from Transformers (BERT) represents a paradigm shift in the encoding of textual information for natural language processing (NLP) tasks [3]. Unlike unidirectional models that process text in a singular direction (left-to-right or right-to-left), BERT’s novelty lies in its ability to pre-train deep bidirectional representations. This characteristic enables the model to access both preceding and following context simultaneously across

all layers. The employment of bidirectional conditioning is critical for sarcasm detection wherein the intended semantic orientation often contradicts the literal meaning, necessitating a comprehensive grasp of context to infer the true sentiment.

BERT’s architecture is grounded in the Transformer model [10], which revolutionized sequence-to-sequence tasks by replacing sequential computation with a self-attention mechanism. This mechanism allows BERT to evaluate and integrate information from all tokens in the dataset regardless of their sequential position, providing a more fluid and enriched mapping of language contexts. Given that sarcasm frequently relies on complex linguistic cues and pragmatic factors, BERT’s ability to interpret such subtleties offers significant advantages in detecting and understanding sarcastic expressions in textual data.

4.2.2 Methodology

To enhance BERT’s understanding of sarcasm, we implemented a two-phase training approach tailored to tackle the subtlety of sarcastic language in online text. Initially, BERT is fine-tuned on a combined dataset that brings together sentences from a variety of sources, ensuring the model encounters a broad spectrum of sarcastic styles and contexts. This stage allows BERT to adjust to the general patterns of sarcasm, preparing it for the nuanced detection required in the next phase.

In the second phase, we focus on domain-specific fine-tuning by introducing the model to the Amazon product review dataset (using all of the features in it, not just the review), which presents a more focused form of sarcasm often seen in customer feedback. This step fine-tunes BERT’s weights, which have already been adjusted for sarcasm, to this particular application. The intent behind this specialized training is to forge a model adept at recognizing sarcasm in the particular environment of e-commerce.

By conducting our training in two stages, we aim to balance a broad understanding of language with a precise capability to detect sarcasm. Each stage is evaluated thoroughly to verify the model’s capability to accurately identify sarcasm in text.

4.2.3 Dataset and Preprocessing

For model training, we developed the *SarcasticSentences-Dataset* to streamline the process of converting raw text into a form suitable for BERT. This dataset class utilizes the BERT tokenizer, applying it to each sentence to generate a sequence of tokens. Critical tokens, ‘[CLS]’ and ‘[SEP]’, are added to the beginning and end of these sequences, respectively, to mark their boundaries as expected by the model.

Given the variability in sentence lengths and BERT’s fixed input size requirement, we established a preprocessing regimen that includes padding shorter sentences and truncating longer ones, enforcing uniformity in sequence length. This approach addresses BERT’s sequence length constraint and ensures all input data is shaped consistently, thereby optimizing it for processing by the model without substantial information loss.

4.2.4 Model Architecture

The cornerstone of our sarcasm detection model is the ‘BertForSequenceClassification’ from the Hugging Face’s ‘transformers’ library, which provides a convenient wrapper around the pre-trained BERT model tailored for sequence classification tasks. This base model comes with a fully connected layer on top for classification purposes, capable of rendering the final prediction for a given input sequence.

To enhance the model’s generalization capabilities and reduce the risk of overfitting, we strategically incorporated a dropout mechanism directly into the classification head. The dropout layers randomly silence a fraction of neurons during the training phase, thereby compelling the model to learn more robust features that do not rely on any specific set of neurons. This is combined with L2 regularization, applied via weight decay in the training optimizer. The regularization technique adds a penalty for large weights to the loss function, encouraging the model to maintain simpler, smaller weights which can improve performance on unseen data.

Moreover, hyperparameters were meticulously chosen to balance the model’s complexity with its predictive power on sarcasm detection. The classification layer utilizes a softmax function to output predicted probabilities, enabling the discernment of the two classes—sarcastic or not sarcastic—by assigning a likelihood to each possible category.

4.2.5 Training Procedure

The training procedure is outlined as a series of steps managed by the `train_epoch` function. Each epoch involves processing batches of data where the model performs forward propagation to generate predictions, calculates the loss using the `CrossEntropyLoss` function appropriate for binary classification, and then adjusts its parameters through back-propagation. An optimizer, specifically AdamW with a defined learning rate and weight decay, then updates the weights to minimize the loss.

The `eval_model` function handles the evaluation phase. It mirrors the training function without performing any weight updates, ensuring the model’s performance is assessed on the validation set without affecting its learned parameters. This phase computes key metrics, such as accuracy, precision, and recall, to evaluate the model’s abil-

ity to discern sarcasm accurately. These metrics offer insights into the model’s proficiency at correctly classifying inputs as either “sarcastic” or “non-sarcastic,” thus providing a quantitative measure of its prediction quality.

4.2.6 Experimental Results

Our model’s performance on the sarcasm detection task was quantitatively evaluated using precision and recall metrics, as detailed in the following results:

Class	Precision	Recall	Accuracy
Sarcasm	0.9138	0.7597	-
Non-Sarcasm	0.9995	0.9999	-
Both	-	-	0.8494

Table 4. Precision and Recall for BERT’s predictions on the validation set of the combined dataset.

The results demonstrate that the model performs exceptionally well on non-sarcastic predictions with near-perfect precision and recall. However, there’s a noticeable difference in the performance metrics for sarcasm, with precision exceeding recall. This indicates that while the model is highly confident in its sarcasm predictions, it tends to miss sarcastic instances more often than it falsely labels regular content as sarcastic.

With these insights, we recognize areas where the model can benefit from further refinement. The next phase of our experimentation aims to use the insights gained from these results to improve the fine-tuning process on the domain-specific Amazon product review dataset, leveraging the learned weights.

4.3. Enhanced BERT-based Sarcasm Detection Model on Amazon Reviews Dataset

Having established a solid basis for sarcasm detection in generic texts, we now turn our attention to the more specialized domain of e-commerce, particularly focusing on Amazon product reviews. These reviews pose unique challenges for sarcasm detection due to the context-specific language, diverse expressions of customer sentiment, and the presence of informal, colloquial language patterns.

4.3.1 Challenges

Several key challenges are anticipated in adapting our model to this specialized domain:

- Differentiating between genuine and sarcastic praise or criticism within product reviews, which may often involve subtle language cues and contextual understanding.

- Addressing varied expressions of sarcasm that are unique to consumer behavior and product-specific references, requiring the model to interpret a complex interplay of sentiments.
- Ensuring robustness against the backdrop of a diverse array of products and user demographics, which may contribute to the variability in the linguistic expression of sarcasm.

The goal is to enhance the model’s adaptability and precision in this particular application while maintaining a balanced performance across both sarcasm and genuine sentiment detection tasks.

4.3.2 Dataset and Preprocessing

Adapting the BERT model to Amazon product reviews required special attention to dataset curation and preprocessing nuances. Our `SarcasticProductReviewDataset` class was designed to tokenize and encode multiple textual features associated with product reviews, such as titles, author names, product details, and the review body, with each text field contributing valuable context for sarcasm prediction.

To address the class imbalance prevalent in the Amazon reviews dataset—where non-sarcastic reviews significantly outnumber their sarcastic counterparts—we implemented a stratified split, thus ensuring each subset maintains a consistent distribution of classes. Additionally, for the training phase, we incorporated a `WeightedRandomSampler` in our `DataLoader`. This stratagem assigned selection weights inversed to class frequencies, enabling the model to train on a balanced representation of both sarcastic and regular reviews. This technique is essential in cultivating a model less biased toward the more dominant class, thereby enhancing its ability to identify sarcasm with better accuracy.

4.3.3 Model Architecture

The architecture of the `ExtendedBertForMultiFeatureClassification` model is an augmentation of the standard BERT sequence classification framework, designed to accommodate the multidimensional nature of Amazon product reviews. Rooted in BERT’s powerful contextual embeddings, we expanded the model’s capacity by incorporating embeddings for textual features beyond the review body: titles, author information, product details, and star ratings.

In our extended model, specific BERT modules are utilized to encode each type of textual information independently, providing a representation enriched with diverse contextual features. These embeddings are then aggregated into a single comprehensive vector representation.

To facilitate the integration of these multi-feature embeddings, we channel the combined output through a fully connected neural network. This network includes layers activated by the ReLU (Rectified Linear Unit) function, which introduces non-linearity, alongside strategically placed dropout layers to mitigate overfitting risks and to foster the robustness of the architecture. The culmination of this processing stream is a final classification layer that computes the logits, signifying the model’s predictions regarding the sarcasm present in a provided review.

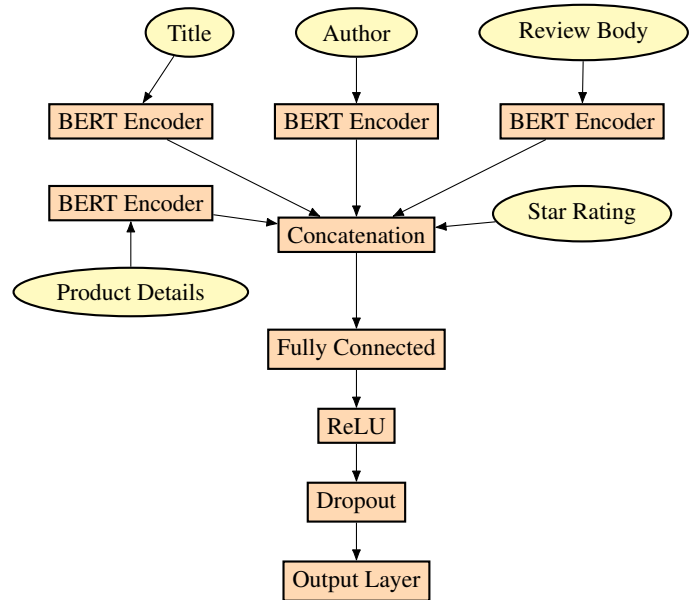


Figure 2. The architecture of the Extended BERT Model for Multi-Feature Classification.

4.3.4 Training Procedure

The training regimen for the Enhanced BERT-based Sarcasm Detection Model aligns with the initial methodology outlined in phases one, leveraging our `train_epoch` and `eval_model` functions. However, the model is further refined with domain-specific nuances of Amazon product reviews. We continue to use `CrossEntropyLoss` as the loss function and employ regularization and dropout techniques to prevent overfitting, complimenting the learned intricacies of sarcasm detection from the preliminary model.

4.3.5 Experimental Results

The enhanced BERT model’s performance on the Amazon Reviews Dataset indicates intricacies in adapting to

domain-specific sarcasm. While the overall accuracy suggests a competent level of sarcasm detection, the results did not markedly surpass those of the preliminary model. The performance of the best model configuration achieved during the training process is as follows:

Class	Precision	Recall	Accuracy
Sarcasm	0.8235	0.6462	-
Non-Sarcasm	0.9995	0.9998	-
Overall	-	-	0.8298

Table 5. Precision, Recall, and Overall Validation Accuracy for the Enhanced BERT Model on the Amazon Reviews Dataset.

The addition of contextual features such as titles and author details did not significantly enhance model performance, potentially pointing to the limitations introduced by the truncation of longer review texts. The truncation may result in the loss of critical information necessary for the BERT model to fully understand the sarcastic or non-sarcastic nature of texts. Additionally, the smaller size of the Amazon Reviews Dataset in comparison to the combined dataset used for the preliminary model may affect the richness of data needed for the model to effectively learn and generalize sarcasm detection. Future work may consider strategies to mitigate information loss from truncation and to utilize a larger, richer dataset for domain-specific training.

4.4. Comparison with Chat-GPT

In order to enhance the depth of our analysis, we conducted a comparative evaluation between our sarcasm detection model and Chat-GPT, primarily focusing on the GPT-3.5-turbo variant developed by OpenAI. The objective of this comparison was to assess the efficacy of our model in comparison to a widely recognized general-purpose language model.

4.4.1 Methodology

Our approach involved querying Chat-GPT with a set of sentences from our dataset, asking it to determine if each was sarcastic. The queries were formulated in a simple way, while we recognize that the design of the prompt was not extensively optimized and may need more modification to improve accuracy. This observation suggests the possibility of making enhancements in further experimental endeavors.

4.4.2 Observations & Thoughts

The preliminary results of Chat-GPT on our sarcasm detection test were unsatisfactory, exhibiting an accuracy rate of merely 59.75%. The actual application of Chat-GPT was hindered by the time-consuming nature of its interaction,

mostly caused by the response latency and rate limiting of the API. This limitation had a notable impact on the usability of Chat-GPT in many circumstances. Furthermore, the responses generated by Chat-GPT occasionally deviated from the intended aim, most likely related to its broad training and reliance on the provided prompts.

Moreover, the financial aspect associated with the utilization of GPT-3.5-turbo proved to be significant, especially considering its rather limited precision as observed in our conducted evaluations. Taking into account these factors, including the processing requirements and concerns regarding response time, raises doubts about the viability of utilizing this approach for sarcasm detection in e-commerce environments on a big scale or in real-time scenarios. Subsequent examinations may be conducted to assess the potentialities of GPT-4; nevertheless, the substantial cost associated with its utilization constitutes a pivotal aspect requiring careful deliberation.

5. What remains to be done

- Feature Integration for BERT Model:** The existing BERT-based model will be improved by integrating supplementary product-specific attributes from the Amazon dataset, including product ratings, names, and other pertinent metadata.
- Comparison between BERT and simpler model using feature selection:** Compare the efficiency of the previously introduced BERT model and an SVM model using features derived from the text as suggested by Jain et al. [6] in their work.
- Refine ChatGPT Approach:** The present approach employed with GPT-3.5-turbo necessitates further refinement. In order to enhance the precision of sarcasm detection within product evaluations, we will investigate more effective prompt designs.
- Evaluate GPT-4 Performance:** Leveraging a subset of our dataset, we plan to test the effectiveness of GPT-4 for sarcasm detection, keeping in mind the associated costs.
- Interpretation and Rephrasing (Time Permitting):** One of the biggest objectives is to make our model comprehend and reformulate sarcastic or amusing reviews into unambiguous and devoid-of-sarcasm textual content. This approach facilitates the identification of genuine customer feelings, even when they are expressed indirectly through sarcasm or comedy.
- Automated Response Generation (Time Permitting):** If feasible within time, our objective is to create a functionality that enables our sarcasm detection

model to autonomously generate responses that consider the tone of a user’s comment or query.

References

- [1] Issa Annamoradnejad and Gohar Zoghi. Colbert: Using bert sentence embedding in parallel neural networks for computational humor, 2020. arXiv:2004.12765, updated Dec 2022. [1](#)
- [2] Center for Information and Language Processing (CILP). Mustard++. https://github.com/cfiltnlp/MUSTARD_Plus_Plus, 2022. Accessed: Jan 15, 2022. [2](#)
- [3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. <https://arxiv.org/abs/1810.04805>, 2018. Proceedings of NAACL-HLT 2019, pages 4171–4186. [3](#)
- [4] Elena Filatova. Sarcasmamazonreviewscorpus. <https://github.com/ef2020/SarcasmAmazonReviewsCorpus>, 2012. Paper: Irony and Sarcasm: Corpus Generation and Analysis Using Crowdsourcing, Proceedings of LREC 2012. [2](#)
- [5] Aishwarya Gupta, Avik Pal, Bholeswar Khurana, Lakshay Tyagi, and Ashutosh Modi. Humor@iitk at semeval-2021 task 7: Large language models for quantifying humor and offensiveness. arXiv:2104.00933, accepted at SemEval 2021. [1](#)
- [6] Sahil Jain, Ashish Ranjan, and Dipali Baviskar. Sarcasm detection in amazon product reviews. *International Journal of Computer Science and Information Technologies*, 2018. [1](#), [2](#), [6](#)
- [7] Rishabh Misra. News-headlines-dataset-for-sarcasm-detection. <https://github.com/rishabhmisra/News-Headlines-Dataset-For-Sarcasm-Detection>, 2019. Accessed: Jul 3, 2019. [1](#)
- [8] Shereen Oraby, Vrindavan Harrison, Lena Reed, Ernesto Hernandez, Ellen Riloff, and Marilyn Walker. 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 (SIGDIAL)*, Los Angeles, California, USA, 2016. Sarcasm Corpus V2: <https://nlds.soe.ucsc.edu/sarcasm2>. [2](#)
- [9] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. A deeper look into sarcastic tweets using deep convolutional neural networks. *Int. Conf. on Computational Linguistics*, 2016. arXiv:1610.08815. [1](#)
- [10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. <https://arxiv.org/abs/1706.03762>, 2017. Proceedings of NIPS 2017, pages 5998–6008. [3](#)
- [11] Hamed Yaghoobian, Hamid R. Arabnia, and Khaled Rasheed. Sarcasm detection: A comparative study, 2021. arXiv:2107.02276. [1](#)
- [12] Yftah Ziser, Elad Kravi, and David Carmel. Humor detection in product question answering systems, 2020. Conference: SIGIR 2020. [1](#)