R^3: Reverse, Retrieve, and Rank for Sarcasm Generation with Commonsense Knowledge

Tuhin Chakrabarty[1,2]*, **Debanjan Ghosh**[3], **Smaranda Muresan**[2,4] **and Nanyun Peng**[1]
[1]Information Sciences Institute, University of Southern California
[2]Department of Computer Science, Columbia University
[3]Educational Testing Service, [4]Data Science Institute, Columbia University

{tc2896, smara}@columbia.edu
dghosh@ets.org, npeng@isi.edu

Abstract

We propose an unsupervised approach for sarcasm generation based on a non-sarcastic input sentence. Our method employs a retrieve-and-edit framework to instantiate two major characteristics of sarcasm: reversal of valence and semantic incongruity with the context, which could include shared commonsense or world knowledge between the speaker and the listener. While prior works on sarcasm generation predominantly focus on context incongruity, we show that combining valence reversal and semantic incongruity based on the commonsense knowledge generates sarcasm of higher quality. Human evaluation shows that our system generates sarcasm better than human annotators *34%* of the time, and better than a reinforced hybrid baseline *90%* of the time.

1 Introduction

Studies have shown that the use of sarcasm or verbal irony, can increase creativity on both the speakers and the addressees (Huang et al., 2015), and can serve different communicative purposes such as evoking humor and diminishing or enhancing critique (Burgers et al., 2012). Thus, developing computational models that generate sarcastic messages could impact many downstream applications, such as better conversational agents and creative or humorous content creation. While most computational work has focused on sarcasm detection (Davidov et al., 2010; González-Ibáñez et al., 2011; Riloff et al., 2013; Ghosh et al., 2015; Joshi et al., 2015b; Muresan et al., 2016; Ghosh and Veale, 2017; Ghosh et al., 2018), research on sarcasm generation is in its infancy (Joshi et al., 2015a; Mishra et al., 2019). Sarcasm generation is a challenging problem since the generated utterance should have at least five characteristics (a.k.a. "sarcasm factors") (Burgers et al., 2012): 1) be evaluative; 2) be based on a reversal of valence between the literal and intended meaning; 3) be based on a semantic incongruity with the context, which can include shared commonsense or world knowledge between the speaker and the addressee; 4) be aimed at some target, and 5) be relevant to the communicative situation in some way. To simplify the problem, we focus on the task of generating a sarcastic utterance starting from a non-sarcastic utterance that conveys the speaker's intended meaning and that is evaluative. Consider the examples at Table 1. Given the literal input "I hate getting sick from fast food" or "I inherited unfavorable genes from my mother", our task is to generate a sarcastic message that would convey that literal (i.e., intended) meaning.

In this simplifying task, we are not concerned with the fifth characteristic, while the first and to some degree, the fourth are specified by the input (literal) utterances.

Literal Input 1	I hate getting sick from fast food.
GenSarc1	I love getting sick from fast food.
GenSarc2	[I love getting sick from fast food.] [Stomach ache is just an additional side effect.]
Human 1	Shout out to the Mc donalds for giving me bad food and making me sick right before work in two hours.
Literal Input 2	I inherited unfavorable genes from my mother.
GenSarc3	I inherited great genes from my mother.
GenSarc4	[I inherited great genes from my mother.] [Ugly goes down to the bone.]
Human 2	Great I inherited all of my mother's GOOD genes

Table 1: Table showing a literal or non sarcastic input sentence and respective sarcastic outputs. GenSarc1 and GenSarc3 simply reverses the valence, while GenSarc2 and GenSarc4 add commonsense context to create incongruity or enhance the humorous effect.

* The research was conducted when the author was at USC/ISI.

Given the lack of "training" data for the sarcasm generation task, we propose a novel *unsupervised approach* that has three main modules guided by the above mentioned sarcasm factors:

1. **Reversal of Valence:** To generate sarcastic utterances that satisfy the second characteristic we identify the evaluative word and use negation or lexical antonyms to generate the sarcastic utterance by reversing the valence (Section 4.1). For example, given, "I **hate** getting sick from fast food" this module will generate "I **love** getting sick from fast food" (GenSarc1 in Table 1).

2. **Retrieval of Commonsense Context:** Adding commonsense context could be crucial to emphasize the semantic incongruity factor (e.g., GenSarc3 vs. GenSarc4 in 1), or could enhance the humorous effect of the generated sarcastic message (e.g., GenSarc1 vs. GenSarc2 in 1).

   We propose an approach where retrieved relevant commonsense context sentences are to be added to the generated sarcastic message. At first, we use a pre-trained language model fine-tuned on the ConceptNet (Speer et al., 2017) called COMET (Bosselut et al., 2019) to generate relevant commonsense knowledge. COMET gives us that, "inherited unfavorable genes from my mother" causes *"to be ugly"* or that "getting sick from fast food" causes *"stomach ache"* (Section 4.2.1). The derived commonsense concept is then used to retrieve relevant sentences — from a corpus — that could be added to the sentence obtained through reversal of valence (e.g., "Stomach ache is just an additional side effect" in Table 1) (Section 4.2.2).

3. **Ranking of Semantic Incongruity:** The previous module generates a list of candidate commonsense contexts. Next, we measure *contradiction* between each of these commonsense contexts and the sentence generated by the reversal of valence approach (module 1) and select the commonsense context that received the highest contradiction score. Finally, we concatenate the selected context to the sentence obtained through reversal of valence. Here, conceptually, contradiction detection is aimed to capture the semantic incongruity between the output of valence reversal and its context. Contradiction scores are obtained from a model trained on the Multi-Genre NLI Corpus (Williams et al., 2018) (Section 4.3).

We test our approach on 150 non sarcastic utterances randomly sampled from two existing data sets. We conduct human evaluation using several criteria: 1) how *sarcastic* is the generated message; 2) how *humorous* it is; 3) how *creative* it is; and 4) how *grammatical* it is. Evaluation via Amazon's Mechanical Turk (MTurk) shows that our system is better 34% of the time compared to humans and 90% of the time compared to a recently published reinforced hybrid baseline (Mishra et al., 2019). We also present a thorough ablation study of several variations of our system demonstrating that incorporating more sarcasm factors (e.g., reversal of valence, commonsense context, and semantic incongruity) assists generate better sarcastic utterances. We make the code and data from our experiments publicly available. [1]

2 Related Work

2.1 Sarcasm Generation

Sarcasm generation is under-explored. Joshi et al. (2015a) proposed *SarcasmBot*, a sarcasm generation system that implements eight rule-based sarcasm generators, each of which generates a certain type of sarcastic expression. Peled and Reichart (2017) introduced a novel task of sarcasm interpretation, defined as the generation of a non-sarcastic utterance conveying the same message as the original sarcastic one. They use supervised machine translation models for the same in presence of parallel data. However, it is impractical to assume the existence of large corpora for training supervised generative models using deep neural nets; we hence resort to unsupervised approaches. Mishra et al. (2019) employed reinforced neural *seq2seq* learning and information retrieval based approaches to generate sarcasm. Their models are trained using only unlabeled non-sarcastic and sarcastic opinions. They generated sarcasm as a disparity between positive sentiment context and negative situational context. We, in contrast, model sarcasm using semantic incongruity with the context which could include shared commonsense or world knowledge.

[1] https://github.com/tuhinjubcse/SarcasmGeneration-ACL2020

2.2 Style Transfer

Prior works looked into *unsupervised* text style/sentiment transfer (Shen et al., 2017; Fu et al., 2017; Li et al., 2018), which transfers a sentence from one style to another without changing the content. This is relevant to the reversal of valence for sarcasm generation. However, these transformations are mainly on the lexical and syntax levels rather than pragmatic level; in contrast, sarcastic utterances often include additional information associated with the context they occur (Regel, 2009), which is beyond text style/sentiment transfer.

2.3 Use of Commonsense for Irony Detection

The study of irony and sarcasm are closely related as sarcasm is defined as, "the use of verbal irony to mock someone or show contempt". Van Hee et al. (2018) first addressed the challenge of modeling implicit or prototypical sentiment in the framework of automatic irony detection. They first manually annotated stereotypical ironic situations (e.g., flight delays) and later addressed the implicit sentiment held towards such situations automatically by using both a lexico-semantic commonsense knowledge base and a data-driven method. They however used it for irony detection, while we are focused on sarcasm generation.[2]

3 Sarcasm Factors Used in Generation

A sarcastic utterance must satisfy the sarcasm factors, i.e., the inherent characteristics of sarcasm (Attardo, 2000; Burgers et al., 2012). In this research, we leverage the use of two particular factors to generate sarcasm. One is the *reversal of valence* and the other is *semantic incongruity with the context*, which could include shared commonsense or world knowledge between the speaker and hearer.

3.1 Reversal of Valence

The first key sarcasm factor is the reversal of valence between the literal and the intended meaning (Burgers et al., 2012). Reversal of valence can be achieved in two ways: when literal meaning of the sarcastic message is positive (e.g., "that is a great outfit" if the outfit is ugly) or when the literal meaning is negative (e.g., "that is an ugly dress" if the dress is really beautiful). Arguably, the former is more likely to appear in sarcastic utterances. As the intended meaning is generally the opposite of its literal meaning in sarcastic utterances (Gibbs, 1986), using lexical antonym of negative sentiment words or negation can serve the purpose to convert a non-sarcastic utterance to its sarcastic version. For example, given a non-sarcastic utterance "zero visibility in fog makes driving **difficult**", one could identify the evaluative negative word *difficult* and replace it with its antonym *easy*, thereby converting the utterance to the sarcastic "zero visibility in fog makes driving **easy**". Likewise, "Drunk driving should be taken seriously" can be converted to its sarcastic counterpart, "Drunk driving should **not** be taken seriously" by using negation. We propose a generation approach that is able to capture the reversal of valence (Section 4.1).

3.2 Semantic Incongruity

The second sarcasm factor semantic incongruity appears between the literal evaluation and the context (e.g., between the positive sentiment words and the negative situation in consideration). In the sarcastic utterance "I love getting sick from fast food" semantic incongruity appears between the positive word "love" and the negative situation "getting sick". However, often, the negative situation is absent in the utterance, thus, additional pragmatic inference is needed to parse its meaning. For example, it might unclear to the listener why "zero visibility in fog makes driving easy" is sarcastic, however the intent of the speaker is to let the listener know that it can cause *"accidents"*. Adding "suffered three cracked ribs in an accident." makes the sarcastic intent clearer while maintaining the acerbic wit of the speaker. In the next section, we propose a novel generation approach that incorporates such relevant commonsense knowledge as context for semantic incongruity (Section 4.2 and Section 4.3).

4 Unsupervised Sarcasm Generation

An overview of the sarcasm generation pipeline is shown in Figure 1. In this section, we detail the three main modules that are designed to instantiate the key sarcasm factors.

4.1 Reversal of Valence

It has been argued that ironic criticism (i.e., mock positive evaluation of negative circumstances) is

[2]While we do not directly model the negative intent in sarcasm, the generated output could lead to sarcastic messages rather than just ironic depending on the initial target given in the non-sarcastic message (E.g a sample generation "Our politicians have everything under control. The nation is in danger of falling into anarchy.")

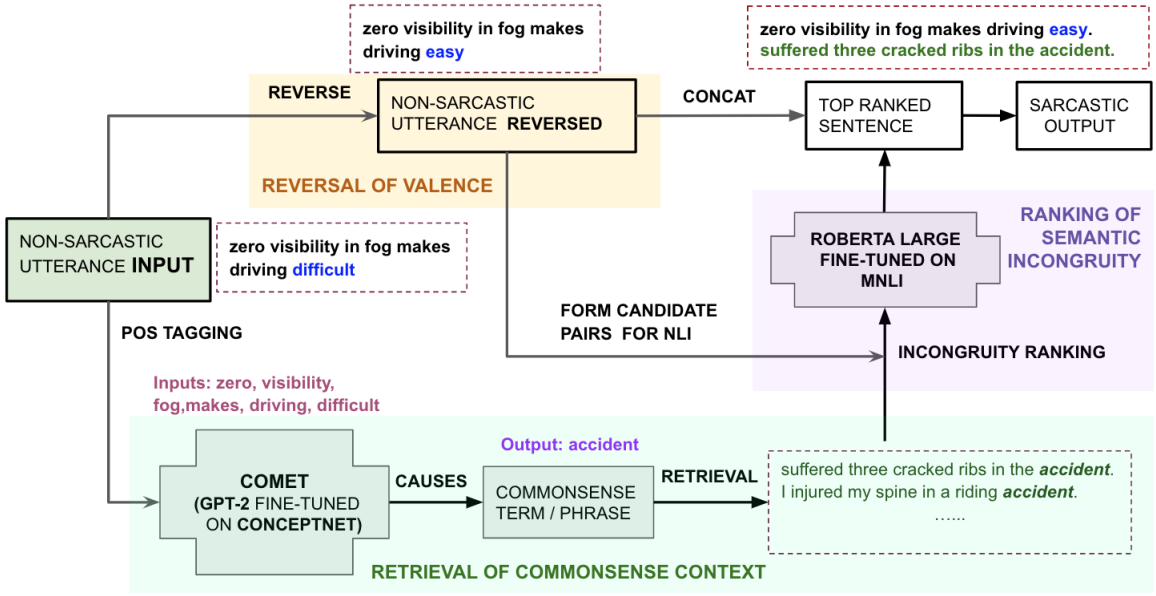

Figure 1: Our complete pipeline for sarcasm generation. The components with highlighted background denote Reversal of Valence, Retrieval of Commonsense Context and Ranking based on Semantic Incongruity respectively

more prototypical than ironic praise (i.e., mock negative evaluation of positive circumstances) (Kreuz and Link, 2002). Sarcasm is a type of verbal irony used to mock or convey contempt. Thus, we mostly encounter positive sentiment towards negative situations in sarcastic messages. This observation is also supported by the sarcasm detection research particularly on social media. Hence, for sarcasm generation, we focus on transforming a literal utterance with negative valence into positive valence.

To implement the reversal of valence, as highlighted in the yellow background in Figure 1, we first identify the evaluative words and replace them with their lexical antonyms using WordNet (Miller, 1995). As we expect the evaluative words to be negative words, we rely on the word level negative scores obtained from SentiWordNet (Esuli and Sebastiani, 2006). In absence of a negative word we check if there is a *not* or words ending with *n't* and remove these words. In case there are both negative words and *not* (or words ending in *n't*), we handle only one of them. The non sarcastic example *"zero visibility in fog makes driving* **difficult***"* shown in Figure 1 serves as our running example. Reversal of valence module generates *"zero visibility in fog makes driving* **easy***"* from the running example.

4.2 Retrieval of Commonsense Context

As discussed before, a straightforward reversal of valence might not generate sarcastic messages that display a clear semantic incongruity, and thus, additional context is needed. We propose an approach to retrieve relevant context for the sarcastic message based on commonsense knowledge. First, we generate commonsense knowledge, e.g., that the running example implies causing accidents (Section 4.2.1). Next, we retrieve candidate context sentences that contain the commonsense concept from a retrieval corpus (Section 4.2.2) and edit for grammatical consistency with the input message (Section 4.2.3).

4.2.1 Commonsense Reasoning

We extract nouns, adjectives, adverbs, and verbs from the non-sarcastic input messages and feed them as input to COMET (Bosselut et al., 2019) model to generate commonsense knowledge (highlighted in green background in Figure 1. COMET is an adaptation framework for constructing commonsense knowledge based on pre-trained language models. It initiates with a pre-trained GPT (Radford et al., 2018) model and fine-tune on commonsense knowledge tuples (in our case, ConceptNet (Speer et al., 2017)). These tuples provide COMET with the knowledge base structure and relations that must be learned, and COMET adapts the representations that the language model learned from the pre-training stage to add novel nodes to the seed knowledge graph. Our work only leverages the **causes** relation. For instance, from our running example, we first remove the stopwords and then extract nouns, adjectives, adverbs, and verbs including the terms *zero*, *visibility*, *fog*,*makes driving*, and *difficult* to feed to COMET as inputs.

In turn, COMET returns the probable causes with their probability scores. For the running example, COMET returns with the highest probability that these terms may cause an **accident** (illustrated in Figure 2). For further details regarding COMET please see Bosselut et al. (2019).

4.2.2 Retrieving Sentences Containing Commonsense Concepts

Once we obtain the most probable output from COMET, the next step is to retrieve sentences containing the commonsense word or phrase from a retrieval corpus. We impose several constraints: (a) the retrieved sentences should contain the commonsense concept at the beginning or at the end; (b) sentence length should be less than twice the number of tokens in the non-sarcastic input to keep a consistency between the length of the non-sarcastic input to its sarcastic version. If none of the commonsense phrase is present in the retrieval corpus, we retrieve sentences containing the nouns within the top most phrase. For example if COMET yields *microwave burger awful* causes the phrase **food to spoil**, and this phrase does not appear in any sentence in the retrieval corpus, we search for *food* and later replace it in the retrieved sentence with *food to spoil*. COMET often returns output with common phrases such as *you to be*, *you to get*, *person will be*, *you have* which we also removed while keeping the main content word (i.e the commonsense concept) We use Sentencedict.com, an online sentence dictionary as the retrieval corpus, where one can find high quality sentences for almost every word obeying the above constraints. [3]

4.2.3 Grammatical Consistency

We first check whether the retrieved sentences are consistent with the non-sarcastic input in terms of the pronouns. If the pronouns are mismatched, then we modify the pronoun of the retrieved sentence to match the pronoun of the non-sarcastic input. In case, the non-sarcastic input does not have any pronoun, but the retrieved sentence does, we simply change that pronoun to "I". For example, if the non-sarcastic input sentence is *"Ignoring texts is literally the worst part of communication."* and the retrieved commonsense sentence is *"**He** has never suffered the torment of rejection."*, we modify the retrieved sentence to *"**I** have never suffered the torment of rejection."* to have consistency among the pronoun use. After correcting the pronouns and proper names (in the same way as pronoun correction), we feed the corrected sentences into the Neural Grammatical Error Corrections System (Zhao et al., 2019) to correct any pronoun or gender specific errors introduced by the replacements.

4.3 Ranking for Semantic Incongruity

After the grammatical error correction, the next step is to select the best context sentence from the retrieved results. Since we expect the context sentences to be incongruous with the sentence generated by the reversal of valence approach (Section 4.1), we rank the context sentences by semantic incongruity scores and select the best candidate.

We borrow the framework of Natural Language Inference (NLI) (Bowman et al., 2015) to measure the incongruity. The Multi-Genre NLI (Williams et al., 2018) covers a range of genres of spoken and written text, and supports a distinctive cross-genre generalization, making it an ideal choice as the NLI Dataset. We first fine-tune RoBERTa-large (Liu et al., 2019), a state-of-the-art pre-trained language model for a 3-way classification (i.e., contradiction, entailment, and neutral) by training on the Multi-NLI dataset. Next, for each retrieved sentence, we treat it as the *premise* and the sentence generated by the reversal of valence as the *hypothesis*, and thus, obtain a contradiction score from the trained model. Finally, scores obtained for the *contradiction* class is used as a proxy for *incongruity* and we select the most incongruent context. Figure 1 shows the region with light purple background as our incongruity ranking module.

4.4 Implementation Details

For commonsense reasoning model, we use the pre-trained COMET model [4] with a greedy decoding of 5 to generate a commonsense phrase and return the topmost that has no lexical overlap with the input. If the generated phrase contains stopwords in the beginning we remove them. For incorporating semantic incongruity, we use the RoBERTa-large model with 355M parameters [5] and fine-tune on MNLI. For grammatical error correction model, we use an open source pre-trained model.[6]

[3] https://sentencedict.com/

[4] https://github.com/atcbosselut/comet-commonsense

[5] https://github.com/pytorch/fairseq/tree/master/examples/roberta

[6] https://github.com/zhawe01/fairseq-gec

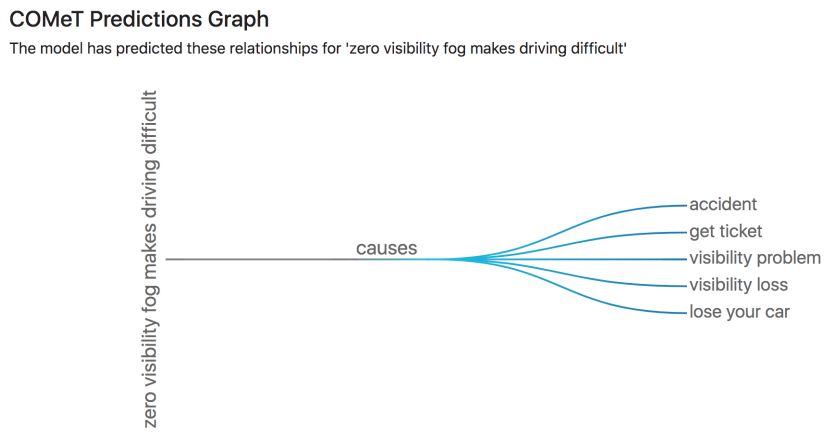

Figure 2: Model predictions from COMET. The edges are sorted by probability

5 Experimental Setup

5.1 Dataset

Ghosh et al. (2020) released a dataset of 4,762 pairs of speakers sarcastic messages and hearers interpretations by conducting a crowdsourcing experiment. Peled and Reichart (2017) introduced a novel dataset of 3,000 sarcastic tweets, each interpreted by five human judges and present a novel task of sarcasm interpretation. Both datasets were collected using the hashtag *#sarcasm* from Twitter. We merge these two datasets and choose non-sarcastic utterances no longer than 15 words. For each literal non-sarcastic utterance we also keep the corresponding gold sarcastic message, which is useful for evaluation and comparison purposes. We randomly select 150 utterances as part of the test set (i.e., five times more than the size of the test data in Mishra et al. (2019)), while assuring such utterances do not contain high lexical overlap. We allow this constraint to evaluate how our method(s) deal with diverse data.

5.2 Systems for Experiment

Here, we benchmark the quality of the generated sarcastic messages by comparing multiple systems.

1. **Full Model (FM)**: This model consists of all the three modules aimed at capturing reversal of valence, commonsense context, and semantic incongruity, respectively.

2. **Reversal of Valence (RV)**: This model relies only on the reversal of valence component.

3. **No Reversal of Valence (NoRV)**: This model only retrieves commonsense context and ranks them based on semantic incongruity.

4. **No Semantic Incongruity (NSI)**: This model relies only on the reversal of valence and retrieval of commonsense context, without ranking based on semantic incongruity. A randomly selected retrieved sentence is used.

5. **MTS2019**: We make use of the model released by Mishra et al. (2019) as it is the state-of-the-art sarcasm generation system.[7]

6. **Human (Gold) Sarcasm**: As described in Section 5.1, we have gold sarcasm created by humans for every non-sarcastic utterance.

5.3 Evaluation Criteria

BLEU (Papineni et al., 2002) is one of the most widely used automatic evaluation metric for generation tasks such as Machine Translation. However, for creative text generation, it is not ideal to expect significant n-gram overlaps between the machine-generated and the gold-standard utterances. Hence, we look to human evaluation. We evaluate on a total of 900 generated utterance since our ablation consisted of six different systems with 150 utterances each.

Sarcasm is often linked with intelligence, creativity, and wit; thus we propose a set of 4 criteria to evaluate the generated output: (1) **Creativity** ("How creative are the utterances ?"), (2) **Sarcasticness** ("How sarcastic are the utterances ?"), (3) **Humour** ("How funny are the sentences ?") (Skalicky and Crossley, 2018), and (4) **Grammaticality** ("How grammatical are the sentences ?"). We design a MTurk task where Turkers were asked to rate outputs from all the six systems. Each

[7] https://github.com/TarunTater/sarcasm_generation

System	Sarcasticness	Creativity	Humor	Grammaticality
State-of-the-art (Mishra et al., 2019)	1.63	1.60	1.50	1.46
Human Generated	**3.57**	3.16	**3.18**	3.98
Reversal of Valence (RV)	3.00	2.80	2.72	**4.29**
No Reversal of Valence (NoRV)	1.79	2.28	2.09	3.91
No Semantic Incongruity (NSI)	3.04	2.99	2.90	3.68
Full Model (FM)	3.23*	**3.24**	3.08*	3.69

Table 2: Average scores for generated sarcasm from all systems as judged by the Turkers. The scale ranges from 1 (*not at all*) to 5 (*very*). For creativity and grammaticality, our models are comparable to human annotation and significantly better than the state-of-the-art ($p < 0.001$). For sarcasticness and humor, the full model is ranked 2nd by a small margin against the human generated message (denoted by *).

Aspect	FM vs Human		FM vs MTS2019	
	win%	lose%	win%	lose%
Sarcasticness	34.0	**55.3**	**90.0**	6.0
Creativity	**48.0**	36.0	**95.3**	4.0
Humor	40.6	**48.0**	**90.0**	4.0
Grammaticality	26.6	**56.6**	**98.0**	1.3

Table 3: Pairwise comparison between the full model (FM) and human generated sarcasm, and between the full model (FM) and the state-of-the-art model in Mishra et al. (2019). Win % (lose %) is the percentage of the FM gets a higher (lower) average score compared to the other method for the 150 human-rated sentences. The rest are ties.

Turker was given the non-sarcastic utterance as well as a group of sarcastic utterances generated by all the six systems (randomly shuffled). Each criteria was rated on a scale from 1 (*not at all*) to 5 (*very*). Finally, each utterance was rated by three individual Turkers. 55, 59, 66, and 60 Turkers attempted the HITs (inter-annotator agreement of 0.59, 0.53, 0.47 and 0.66 for tasks of creativity, sarcasticness, humour and grammaticality, respectively using Spearman's correlation coefficient).

6 Experimental Results

6.1 Quantitative Scores

Table 2 presents the scores for the above mentioned metrics of different systems averaged over 150 test utterances. Our full model along with the variations that ablated some components all improve over the state-of-the-art (Mishra et al., 2019) on all the criteria. The ablation in Table 2 shows that our full model is superior to individual module in terms of sarcasticness, creativity and humor. For grammaticality, we observe that the Turkers scored shorter sentences higher (e.g., RV), which also explains why NoRV model received a higher score than the full model. It otherwise performed worse than all the other variations.

In terms of creativity, our full model attains the highest average scores over all the other models including sarcastic utterances composed by human. For grammaticality, the reversal of valence model is the best, even better than human generated ones. The performance of the full model is the second best in terms of the sarcasticness and humor, only slightly worse than human-generated sarcasm, showing the effectiveness of our approach that captures various factors of sarcasm.

6.2 Pairwise game between Full Model, State-of-the-art and Humans

Table 3 displays the pairwise comparisons between the full model (FM) and human generated sarcasm, and FM and Mishra et al. (2019), respectively. Given a pair of inputs, we decide win/lose/tie by comparing the average scores (over three Turkers) of both outputs. We see that FM dominates Mishra et al. (2019) on all the metrics and human-generated sarcasm on the creativity metric. For sarcasticness, although humans are better, the FM model still has a 34% winning rate.

6.3 Ablation Study

We focus our ablation study on the metric of sarcasticness, as we consider this as the chief criteria in success of generating sarcasm. As shown in Figure 3, our best model (FM) outperforms individual ablation module. We filtered out 60 examples from the 150 with no ties. The ablation component employing just *Reversal of Valence* is second best for sarcasticness according to Figure 3.

Further, to understand the extent to which ranking the retrieved sentence based on the degree of incongruity helped generate better sarcasm, we took the outputs from FM and NSI for comparisons. Out

Non Sarcastic	System	Sarcasm	S	C	H	G
I inherited unfavorable genes from my mother.	FM	I inherited great genes from my mother. **Ugly** goes down to the bone.	**5.0**	4.0	**3.6**	3.6
	RV	I inherited great genes from my mother.	3.0	2.6	2.0	2.3
	NoRV	**Ugly** goes down to the bone.	3.0	2.6	3.0	**4.0**
	NSI	I inherited great genes from my mother. She makes me feel dowdy and **ugly**.	2.6	3.6	3.0	**4.0**
	MTS2019	Butch tagging bullies apc seymour good temper good mentor.	1.3	1.0	1.3	2.0
	Human	Great I inherited all of my mother's GOOD genes	2.3	**4.3**	2.0	2.6
It is not fun to date a drug addict.	FM	It is fun to date a drug addict. Spent the night in a police cell after his **arrest**.	4.3	**5.0**	**4.6**	**5.0**
	RV	It is fun to date a drug addict.	**5.0**	2.3	2.0	4.6
	NoRV	Spent the night in a police cell after his **arrest**.	1.0	1.0	2.0	2.6
	NSI	It is fun to date a drug addict. The feds completely screwed up the **arrest**.	3.3	4.3	2.0	2.6
	MTS2019	Butch is a powerful addict in gente he is an optimist great fun.	2.6	2.0	1.0	1.3
	Human	Dating a drug addict .. Wouldn't that be fun.	3.0	1.6	2.6	4.0
I hate getting sick from fast food.	FM	I love getting sick from fast food. **Stomach ache** is just an additional side effect.	3.3	3.6	**5.0**	3.6
	RV	I love getting sick from fast food.	3.3	2.6	3.6	**5.0**
	NoRV	**Stomach ache** is just an additional side effect.	1.3	2.6	3.6	3.3
	NSI	I love getting sick from fast food. I ate too much and got a terrible **stomach ache**.	2.3	3.3	4.3	**5.0**
	MTS2019	I hate love sick to ikes sword lowest **** giving stains giving stains on printers making pound accidents work bikinis in	1.0	1.3	1.3	1.0
	Human	Shout out to the mcdonalds for giving me bad food and making me sick right before work in two hours.	**4.0**	**4.3**	4.0	4.3
Burnt popcorn is gross.	FM	Burnt popcorn is lovely. The smell made me want to **vomit.**	**4.6**	3.0	3.3	**5.0**
	RV	Burnt popcorn is lovely.	4.0	2.0	3.6	**5.0**
	NoRV	The smell made me want to **vomit.**	1.0	2.0	3.6	4.6
	NSI	Burnt popcorn is lovely. Hold the bag in case I **vomit**.	4.3	2.3	4.3	**5.0**
	MTS2019	reggae burnt popcorn lol .	2.3	1.3	2.0	1.0
	Human	Gotta love the smell of burnt microwave popcorn.	3.3	**3.3**	**4.0**	4.0

Table 4: Examples of generated outputs from different systems. S, C, H, G represent Sarcasticness, Creativity, Humor and Grammaticality. Text in bolded black represents the commonsense word/phrase obtained from COMET given the non-sarcastic utterance.

of the 150 utterances, 119 times there wasn't a tie. Our best model (FM) wins 66% of the time while the NSI model wins 34% of the cases.

7 Qualitative Analysis

Table 4 demonstrates several generation outputs from different modules associated with human ratings for different criteria. We notice that often one of our modules generate better sarcasm than humans. For instance for the first and the second example in Table 4, all of FM, RV and NSI are better than human generated sarcasm. In general, the generations from the FM model are more humorous which is also an useful criteria to evaluate sarcasm besides sarcasticness (Skalicky and Crossley, 2018).

We also observe that Turkers consistently rated generations from the FM model more sarcastic than the NSI model suggesting that there is a correlation between human scores of sarcasticness and incongruity. To support this observation, we took the contradiction scores from the RoBERTa model for both best ranked retrieved sentences (FM) and the randomly selected retrieved sentences (NSI). We then computed a correlation between the sarcasticness scores given by the humans and the automatic contradiction scores for both the best ranked retrieved sentences (FM) and the randomly selected

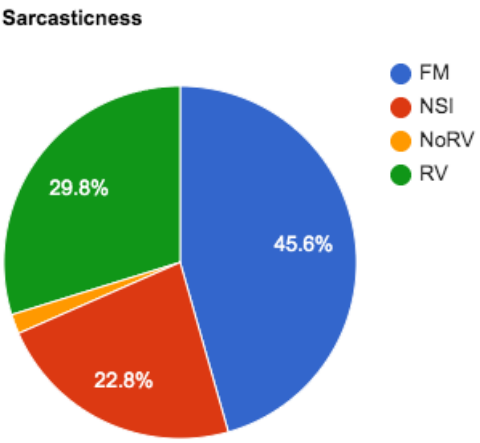

Figure 3: Pie chart comparing the success rate of all the variations of our model.

retrieved sentences (NSI). For FM model we obtain a higher Pearson correlation coefficient compared to for NSI suggesting the important role of incongruency for sarcasm.

7.1 Limitations

While our best model combining different sarcasm factors does outperform the system with individual factors, there are often exceptions. We notice, in few cases, the simple reversal of valence (RV) strategy is enough to generate sarcasm. For instance, for the literal input "It is not fun to date a drug addict", just removing the negation word leads to a full score on sarcasticness without the additional commonsense module. Future work would include building a model that can decide whether just the RV strategy is sufficient or if we need to attach an incongruent commonsense context to it.

Although incorporating incongruity ranking is useful, there are several cases when a randomly retrieved message may obtain better sarcasticness score. Table 5 presents such an example. Even though the retrieved message "Please stop whirling me round; it makes me feel sick." scores lower than "The very thought of it makes me feel sick.", in terms of incongruity with respect to "I love being put in the hospital for dehydration", the former received a higher sarcasticness score that suggest the incongruity scores obtained from NLI are not perfect.

The ordering of the commonsense context and the valence reversed sentence is predetermined in our generation. Specifically, we always append the retrieved commonsense context after the valence reversed output. Changing the order can sometimes

NSI	I love being put in the hospital for dehydration. Please stop whirling me round; it makes me feel **sick**.
FM	I love being put in the hospital for dehydration. The very thought of it makes me feel **sick**.

Table 5: Sarcastic Generation from (FM) and (NSI) where NSI scores higher for sacrasticness

make the sarcasm better and more humorous. The reason for our current ordering choice is that we always treat the valence reversed version as *hypothesis* and the commonsense retrieved sentence as *premise* for the NLI model. We attempted reversing the order in preliminary experiments but received poor scores from the entailment model. In future, we would like to generate more diverse sarcasm that are not tied to a fixed pattern.

Finally, the generations are dependent on COMET and thus the quality will be governed by the accuracy of the COMET model.

8 Conclusion

We address the problem of unsupervised sarcasm generation that models several sarcasm factors including reversal of valence and semantic incongruity with the context. The key contribution of our approach is the modeling of commonsense knowledge in a retrieve-and-edit generation framework. A human-based evaluation based on four criteria shows that our generation approach significantly outperforms a state-of-the-art model. Compared with human generated sarcasm, our model shows promise particularly for creativity, humor and sarcasticness, but less for grammaticality. A bigger challenge in sarcasm generation and more generally, creative text generation, is to capture the difference between creativity (novel but well-formed material) and nonsense (ill-formed material). Language models conflate the two, so developing methods that are nuanced enough to recognize this difference is key to future progress.

Acknowledgments

This work is funded by the Contract W911NF-15-1-0543 with the US Defense Advanced Research Projects Agency (DARPA). The authors would like to thank Christopher Hidey, Anusha Bala, Christopher Robert Kedzie for useful discussions. The authors also thank members of PLUSLab at the University Of Southern California and the anonymous reviewers for helpful comments.

References

Salvatore Attardo. 2000. Irony markers and functions: Towards a goal-oriented theory of irony and its processing. *Rask*, 12(1):3–20.

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.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Christian Burgers, Margot Van Mulken, and Peter Jan Schellens. 2012. Verbal irony differences in usage across written genres. *Journal of Language and Social Psychology*, 31(3):290–310.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *In Proceedings of the Fourteenth Conference on Computational Natural Language Learning. CoNLL.*

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *LREC*, volume 6, pages 417–422. Citeseer.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2017. Style transfer in text: Exploration and evaluation. In *Proceedings of AAAI*.

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

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.

Debanjan Ghosh, Elena Musi, Kartikeya Upasani, and Smaranda Muresan. 2020. Interpreting verbal irony: Linguistic strategies and the connection to the type of semantic incongruity. *Proceedings of the Society for Computation in Linguistics*, 3(9):76–87.

Raymond W Gibbs. 1986. On the psycholinguistics of sarcasm. *Journal of Experimental Psychology: General*, 115(1):3.

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: Short Papers - Volume 2*, HLT '11, pages 581–586, Stroudsburg, PA, USA. Association for Computational Linguistics.

Li Huang, Francesca Gino, and Adam Galinsky. 2015. The highest form of intelligence: Sarcasm increases creativity for both expressers and recipients. *Organizational Behavior and Human Decision Processes*, 131.

Aditya Joshi, Anoop Kunchukuttan, Pushpak Bhattacharyya, and Mark James Carman. 2015a. Sarcasmbot: An open-source sarcasm-generation module for chatbots.

Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015b. Harnessing context incongruity for sarcasm detection. In *ACL*, pages 757–762.

Roger J Kreuz and Kristen E Link. 2002. Asymmetries in the use of verbal irony. *Journal of Language and Social Psychology*, 21(2):127–143.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: a simple approach to sentiment and style transfer. 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 1865–1874.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Abhijit Mishra, Tarun Tater, and Karthik Sankaranarayanan. 2019. A modular architecture for unsupervised sarcasm generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6146–6155, Hong Kong, China. Association for Computational Linguistics.

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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Lotem Peled and Roi Reichart. 2017. Sarcasm SIGN: Interpreting sarcasm with sentiment based monolingual machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1690–1700, Vancouver, Canada. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Stefanie Regel. 2009. *The comprehension of figurative language: electrophysiological evidence on the processing of irony*. Ph.D. thesis, Max Planck Institute for Human Cognitive and Brain Sciences Leipzig.

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.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in neural information processing systems*, pages 6830–6841.

Stephen Skalicky and Scott Crossley. 2018. Linguistic features of sarcasm and metaphor production quality. In *Proceedings of the Workshop on Figurative Language Processing*, pages 7–16.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. *In Thirty-First AAAI Conference on Artificial Intelligence.*

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. We usually dont like going to the dentist: Using common sense to detect irony on twitter. *Computational Linguistics*, 44(4):793–832.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122. Association for Computational Linguistics.

Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. 2019. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 156–165.