PINEAPPLE: Personifying INanimate Entities by Acquiring Parallel Personification Data for Learning Enhanced Generation

Sedrick Scott Keh [1], Kevin Lu [2], Varun Gangal*[1], Steven Y. Feng*[3], Harsh Jhamtani [1], Malihe Alikhani [4], Eduard Hovy [1]
[1]Carnegie Mellon University, [2]University of Waterloo,
[3]Stanford University, [4]University of Pittsburgh
{skeh,vgangal,jharsh,hovy}@cs.cmu.edu, syfeng@stanford.edu
kevin.lu1@uwaterloo.ca, malihe@pitt.edu

Abstract

A personification is a figure of speech that endows inanimate entities with properties and actions typically seen as requiring animacy. In this paper, we explore the task of personification generation. To this end, we propose **PINEAPPLE**: **P**ersonifying **IN**animate **E**ntities by **A**cquiring **P**arallel **P**ersonification Data for **L**earning **E**nhanced Generation. We curate a corpus of personifications called *PersonifCorp*, together with automatically generated de-personified literalizations of these personifications. We demonstrate the usefulness of this parallel corpus by training a seq2seq model to personify a given literal input. Both automatic and human evaluations show that fine-tuning with *PersonifCorp* leads to significant gains in personification-related qualities such as animacy and interestingness. A detailed qualitative analysis also highlights key strengths and imperfections of **PINEAPPLE** over baselines, demonstrating a strong ability to generate diverse and creative personifications that enhance the overall appeal of a sentence. [1]

1 Introduction

Personification is the attribution of animate actions or characteristics to an entity that is inherently inanimate. Consider, for example, the sentence *"The **stars danced playfully** in the moonlit sky."* Here, the vibrance of the stars (something inanimate) is being likened to dancing playfully, which is a distinctly animate action. By allowing readers to construct clearer mental images, personifications enhance the creativity of a piece of text (Bloomfield, 1980; Dorst, 2011; Flannery, 2016).

Being able to automatically identify and generate personifications is important for multiple reasons. First, humans naturally use personifications when communicating. When we say something like *"My phone has died,"* or *"My car is not cooperating,"* to a dialogue system, it is important that the dialogue system understands the intended meaning behind these personifications. If these systems interpret personifications literally, they may fail in several downstream tasks (e.g. classification) since their understanding is incorrect. Being able to generate personifications also allows dialogue agents and language models to be more creative and generate more figurative sentences. Personification generation has additional applications such as AI-assisted creative writing, since machine-generated figures of speech have been shown to enhance the interestingness of written text (Chakrabarty et al., 2021).

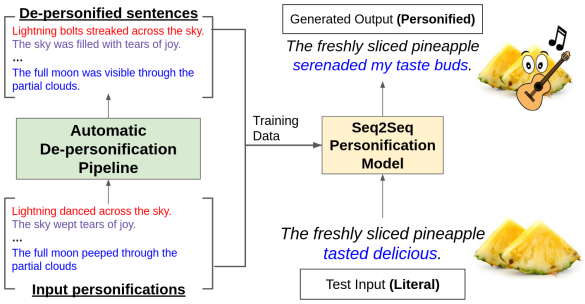

Figure 1: Overall **PINEAPPLE** model pipeline. The left part of the diagram shows the corpus creation process, while the right part of the diagram shows the training and generation process.

Despite previous success in generating other figures of speech such as similes (Chakrabarty et al., 2020), metaphors (Stowe et al., 2021), hyperboles (Troiano et al., 2018), irony (Van Hee et al., 2018), and sarcasm (Hazarika et al., 2018; Jaiswal, 2020), personification generation is relatively underexplored. One key challenge is that personifications do not have an explicit syntactic structure unlike similes which use *'like'* or *'as'*. They are also not as loosely-defined as metaphors. Rather, a personification requires identifying an inanimate subject together with actions or descriptions which are

* Equal contribution by Varun and Steven
[1]Data and code can be found at https://github.com/sedrickkeh/PINEAPPLE

Proceedings of the 29th International Conference on Computational Linguistics, pages 6270–6284
October 12–17, 2022.

commonly used on animate subjects. These steps are challenging and require our models to understand commonsense concepts including animacy.

In line with exploring the task of personification generation, we present three main contributions: (1) We curate a dataset, *PersonifCorp*, of diverse personification examples from various sources. (2) We propose a method called **PINEAPPLE** to automatically de-personify personifications and create a parallel corpus of personification data along with their literalizations. (3) Given our parallel corpus, we train a seq2seq model to personify given text. We conduct automatic and human evaluation and qualitative analysis of the generated outputs.

2 Datasets

We curate a dataset called *PersonifCorp* of 511 personifications, with 236 coming from a publicly available open-sourced list[2] and 275 manually-filtered personifications extracted from the Deja dataset (Chen et al., 2015). The Deja dataset is an image-captioning dataset containing a "figurative" subset of size 6000, of which 4.1% of the captions are labelled as personifications. We extract these personifications and combine them with our existing list to form the final *PersonifCorp* dataset.

We also note that although it is possible to further expand this dataset (e.g. by ad hoc searching for miscellaneous sites and examples online), we ultimately decide against this after performing an initial investigation. When we attempted to look for additional examples, we found that many of the new examples we found were near-duplicates of existing personifications already in our list. In addition, ad hoc searching can give at most a few hundred examples, which will lead to very incremental gains in performance. This is impractical if we want to collect a large-scale dataset. We hence decided to restrict ourselves to sentences from reasonably well-vetted, already existing corpora from *CL prior art or officially released data from sources like Kaggle/SemEval shared tasks.

2.1 Characterizing Personifications

We define the *elements of personification*, an analogue to what was previously done for similes (Niculae and Danescu-Niculescu-Mizil, 2014; Chakrabarty et al., 2020). While similes could be decomposed into very granular structures and well-defined elements, the unstructured nature of personifications prevents us from directly defining such fine-grained elements for personifications. Rather, we define two main high-level elements, the TOPIC (a noun phrase that acts as logical subject) and the ATTRIBUTE (the distinctly animate action or characteristic that is being ascribed to the TOPIC). Figure 2 shows examples of how these TOPICS and ATTRIBUTES can relate to each other.

2.2 Automatic Parallel Corpus Construction

In order to train a seq2seq model to generate high-quality personifications, we need pairs of personifications along with their corresponding literalizations. However, the literalization process may take several human-hours, which is impractical for large datasets. We therefore propose **PINEAPPLE**, a three-stage automatic de-personification process, where we first identify all valid TOPIC-ATTRIBUTE pairs, then generate multiple candidates to replace the ATTRIBUTE of each TOPIC. Lastly, we select the most appropriate candidate in terms of animacy, fluency, and meaning preservation. These steps are further detailed individually below:

TOPIC-ATTRIBUTE Extraction. To identify the TOPICS and ATTRIBUTES, we consider the dependency parse tree of a sentence and the part-of-speech (POS) tags of each of its words. Given the tree, we extract all the nouns/pronouns which have edges pointing into it with the *nominal subject* label, together with the corresponding parent nodes. For instance, in the sentence *"The stars danced in the night sky"*, the word *'danced'* is a parent of the word *'stars'*, with the *nominal subject* edge relationship. We can thus identify *'stars'* as the TOPIC and *'danced'* as the ATTRIBUTE. In more complex scenarios, we may need to perform some additional merging to deal with compound multi-word TOPICS and ATTRIBUTES, as well as any additional modifiers. More specifically, using the POS tags, we identify all words tagged as *negation modifiers*, *possession modifiers*, *nominal modifiers*, *adjectival complements*, and *objects of prepositions*, and words tagged as determiners and parts of compound phrases.[3] After extracting these nodes, they are iteratively merged with their parents in the dependency parse tree, and the merging process is performed repeatedly until no more merges are possible. The final TOPIC-ATTRIBUTE pairs

[2] https://www.kaggle.com/datasets/varchitalalwani/figure-of-speech

[3] The spaCy library was used to extract the dependency tree and POS tags.

ATTRIBUTE Type	Example
Noun	The planet earth is our mother.
Verb	My alarm clock yells at me to get out of bed every morning.
Adjective	Justice is blind and, at times, deaf.

Figure 2: Examples of different types of personification ATTRIBUTES (TOPICS in red and ATTRIBUTES in blue).

are then identified using the *nominal subject* edge relationship as previously described. Examples of the merging process can be found in Appendix A.1.

Candidate Generation. Once the TOPIC-ATTRIBUTE pairs have been identified, we then determine which TOPICS are inanimate. To achieve this, we need some type of commonsense notion of what constitues animacy. We use COMET (Bosselut et al., 2019) to tap into the commonsense knowledge present in large-scale knowledge graphs such as ConceptNet (Speer et al., 2017). Although ConceptNet has no explicit notion of animacy, it has certain edge relations that we can leverage to design a proxy metric. More specifically, we use the *IsA* relation to design a custom *IsAPerson* animacy metric. If the TOPIC of our sentence refers to an animate entity, then we expect its *IsA* relation score with the word *'human'* to be relatively low.[4] The *IsAPerson* metric is hence defined as follows: given a TOPIC, we compute and average its *IsA* scores to various words that are synonymous or very closely related to *'human'*, such as *'person'*, *'man'*, and *'woman'*. We call this set of *'human'*-related words the HUMANSET. The construction and full list of words in the HUMANSET can be found in Appendix A.2. The average of these ConceptNet scores is then our final *IsAPerson* animacy score.

Phrases whose *IsAPerson* animacy score exceeds a certain threshold [5] are considered animate; otherwise, they are considered inanimate. Since our goal is to de-personify a sentence, we can safely discard all the animate TOPICS, as these need no further de-personification. Rather, we focus on the inanimate TOPICS because the segment we want to de-personify most likely occurs in the TOPIC-ATTRIBUTE pairs whose TOPIC is inanimate. Once we identify all such inanimate TOPIC-ATTRIBUTE pairs, we mask out the ATTRIBUTE of each of them with <mask>, then use a pre-trained BART model (Lewis et al., 2020) to generate the top $k = 10$ candidates for each mask using beam search with a beam size of 10. The goal of this process is to replace a possibly animate action/characteristic with candidates that are inanimate.

Candidate Selection. Given $k = 10$ candidate replacement ATTRIBUTES, we now select the most ideal replacement based on three metrics: animacy, fluency, and meaning preservation.

1. Animacy – We want the replacement ATTRIBUTE to be inanimate; otherwise we would just be replacing an animate ATTRIBUTE with another animate ATTRIBUTE. We define the animacy of a TOPIC-ATTRIBUTE pair as difference between the affinity for a human ($\mathcal{A}_{human,\text{ATT}}$) to do/possess the ATTRIBUTE, and the affinity for the given TOPIC ($\mathcal{A}_{\text{TOPIC,ATT}}$) to do/possess the ATTRIBUTE. We use COMET's ConceptNet relations to compute these affinities; specifically, we use the *CapableOf* relation. To approximate $\mathcal{A}_{human,\text{ATT}}$, we compute the average *CapableOf* score between the given ATTRIBUTE and all words in our previously defined HUMANSET. To compute $\mathcal{A}_{\text{TOPIC,ATT}}$, we compute the *CapableOf* score between the TOPIC and its ATTRIBUTE. The final animacy score of a TOPIC-ATTRIBUTE pair is defined as the difference $\mathcal{A}_{human,\text{ATT}} - \mathcal{A}_{\text{TOPIC,ATT}}$. If there are multiple TOPIC-ATTRIBUTE pairs, we consider the average animacy of all pairs.

2. Fluency – The de-personified sentences should be grammatically correct and sound natural. To measure for fluency, we use BART's generation scores (i.e. sum of individual token logits in the generated output).

3. Meaning Preservation – It is important that the de-personified sentence does not stray too far from the meaning of the original personification. We use BERTScore (Zhang* et al., 2020) between the de-personified and original sentences to measure meaning preservation.

We design a composite scoring metric comprised of the aggregate scores from these 3 metrics. Due to scaling differences, we consider the log of the animacy score. To account for the fact that lower animacy scores imply less animate TOPIC-ATTRIBUTE pairs (which is desirable in de-personification), we

[4] For the COMET ConceptNet graph, lower scores correspond to better matches.

[5] We use a threshold of 7.0 for the *IsAPerson* animacy metric. *IsAPerson* scores < 7.0 are considered animate, while scores ≥ 7.0 are considered inanimate. More details regarding the selection of this threshold can be found in Appendix A.3.

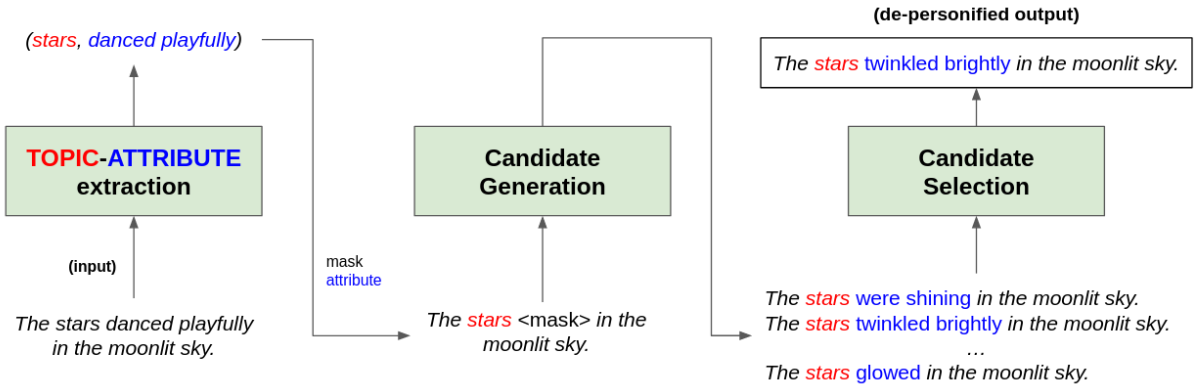

Figure 3: Overview of the **PINEAPPLE** de-personification pipeline.

Original Personification	Result After De-Personifying
How far that little candle throws its beams!	How far that little candle can spread its beams!
A book is a fragile creature, it suffers the wear of time, it fears rodents, the elements and clumsy hands.	A book is fragile, it can break from the wear of time, it can be eaten by rodents, the elements and clumsy hands.
The camera loves her since she is so pretty.	The camera is always on her since she is so pretty.
Any trust I had for him walked right out the door.	Any trust I had for him had gone right out the door.
The full moon peeped through partial clouds.	The full moon was visible through partial clouds.
Opportunity was knocking at her door.	Opportunity was knocking at her door.
The killing moon will come too soon.	The killing moon will be here too soon.

Table 1: Example outputs of the **PINEAPPLE** de-personification pipeline. The ATTRIBUTES are highlighted in blue for both the original personifications, as well as the de-personified output sentences. The last two rows contain negative examples where the process does not successfully de-personify the input.

take the negative of the animacy. More precisely, we define our candidate score S_i for candidate i as

$$S_i = \alpha \cdot (-\log(S_{anim.})) + \beta \cdot S_{flue.} + \gamma \cdot S_{mean.}$$

where α, β, γ are parameters. [6]

Once S_i is computed for all candidates, we select the candidate with the highest composite score as our final de-personified sentence. A diagram of the entire **PINEAPPLE** pipeline is shown in Figure 3, and example outputs can be found in Table 1.

2.3 Test Data Construction

While automatically generated pairs of personifications and literal de-personifications may greatly assist with training, these may not necessarily be accurate for testing. Rather, it would be more ideal during testing if we have ground-truth human-annotated data. To mimic our task at hand, we gather a list of non-personified English sentences.[7] We then select two annotators who are native English speakers currently enrolled in a university with English as a medium of instruction. These annotators were instructed to manually personify these sentences to create ground-truth reference personifications. The final *PersonifCorp* test split has 72 literal + personified sentence pairs.

3 Experimental Setup

3.1 Methods

Below we outline the three models we consider, with two of them being naive baselines (COMET and Baseline-BART) that we simply use on *PersonifCorp*'s test set, and the third (Finetuned-BART) being our proposed model trained on *PersonifCorp*.

1. **COMET**: We extract the TOPIC-ATTRIBUTE pairs and identify the inanimate TOPICS using the methods detailed in §2.2. Instead of generating candidate replacements using BART like in §2.2, we generate candidates by considering the top $k = 10$ results for a given TOPIC using COMET's ConceptNet *IsCapable* relation (if the original ATTRIBUTE is a verb) or *HasProperty* relation (if adjective or adverb). To incorporate a notion of animacy, we use the previously defined ATTRIBUTE animacy $\mathcal{A}_{human,\text{ATT}}$ and select the candidate

[6] We use $\alpha = 1$, $\beta = 1$, $\gamma = 1$. Details about the tuning and selection of α, β, γ can be found in Appendix A.3.

[7] https://github.com/tuhinjubcse/SimileGeneration-EMNLP2020#set-up-data-processing-for-simile

with highest animacy as our final replacement.

2. **Baseline-BART (BL-BART)**: We imitate the process outlined for the COMET baseline, except we use a pretrained BART model to generate the candidates instead of using COMET. All other steps (TOPIC-ATTRIBUTE extraction and candidate selection) remain the same.

3. **PINEAPPLE-BART (PA-BART)**: We fine-tune a BART model by supplying the *PersonifCorp* train split literal de-personified sentences (from the **PINEAPPLE** pipeline) as inputs, and the original ground-truth personifications as target outputs. This is trained as a seq2seq task. During generation, we use beam search. Further details are outlined in §3.3.

3.2 Evaluation

We consider both automatic evaluation metrics (§3.2.1) and human evaluation (§3.2.2).

3.2.1 Automatic Evaluation

For each model in §3.1, we evaluate its generated outputs on *PersonifCorp*'s test split using each of the following automatic evaluation metrics:

1. **BLEU** (Papineni et al., 2002): We use BLEU to ensure that the generations do not greatly differ from the inputs. We compute the BLEU score of each generated output with the literal inputs (for meaning preservation), as well as the ground-truth reference personifications.

2. **BERTScore** (Zhang et al., 2019): BERTScore measures how semantically related two sentences are, and is generally more robust than BLEU. We compute the BERTScore of each generated output with the inputs, as well as the ground-truth reference personifications.

3. **Fluency**: To approximately measure the fluency of a sentence, we use generation (log-perplexity) losses of each output using the GPT-2 language model (Radford et al., 2019).

4. **Animacy**: We are interested in how *personified* the generated output is. We use the same animacy metric used for candidate selection in §2, which is a combination of how animate the ATTRIBUTE is, as well as how inanimate the TOPIC is. More precisely, this is defined as $\mathcal{A}_{human,\text{ATT}} - \mathcal{A}_{\text{TOPIC,ATT}}$, where the $\mathcal{A}$ animacy scores are previously defined in §2.

3.2.2 Human Evaluation

The human evaluation was conducted using paid annotators on Amazon Mechanical Turk (AMT). Annotators were from Anglophone countries with $> 97\%$ approval rate.[8] Each test example was evaluated by exactly 2 annotators. For each test example, we first generate outputs using each of the methods outlined in §3.1. Corresponding to this test instance, we then create an AMT task page (a HIT), presenting the input literal sentence and each of the method outputs (in randomized order) for annotation along five aspects of text quality.

Specifically, annotation was elicited for the following metrics: **(1) Personificationhood** (*"To what extent does the new sentence contain a personification?"*), **(2) Appropriateness** (*"Do the personified nouns, verbs, adjectives, adverbs sound mutually coherent and natural?"*), **(3) Fluency** (*"Does it sound like good English with good grammar?"*), **(4) Interestingness** (*"How interesting and creative a rephrasing of the original sentence is the personified sentence?"*), and **(5) Meaning Preservation** (*"Do the entities, their actions, interactions, and the events appear and relate to each other in the same way as in the original sentence?"*). Each metric was scored on a Likert scale, with 1 being the lowest and 5 being the highest.

For *Interestingness*, we observed poor agreement scores amongst the AMT annotators.[9] Hence, for this aspect, we instead used a curated group of known, in-person annotators: a cohort of three native English-speaking students from an American university. Amongst these annotators, we observe a considerably higher agreement, with a Krippendorff α value of 0.5897. For selecting this cohort from a slightly larger pool of candidates, we assessed their performance on a short qualification test of basic English literary skills and knowhow. The final cohort chosen each scored 85% or higher on this test. Further details are in Appendix B.3.

3.3 Implementation Details

The *PersonifCorp* training corpus was randomly split into a training and validation split with an 80-20 ratio. We fine-tune a BART-base model with 139M parameters using a learning rate of 2e-5 and a batch size of 4. Training was done for 20 epochs and 400 warmup steps, and model/epoch selection

[8] More details about the human eval are in Appendix B.1.

[9] Further details on inter-annotator agreement scores can be found in Appendix B.2.

was performed based on the lowest validation loss. For generating the outputs, decoding was done using beam search with a beam size of 10. Additional details can be found in Appendix C.

4 Results and Analysis

4.1 Automatic Evaluation Results

Table 2 reports the automatic evaluation results for each of the metrics detailed in §3.2.1. We observe that our PA-BART model performs best across all automatic metrics except for fluency, where BL-BART performs best. The difference in performance is most significant in the *Animacy* metric, which is the key metric that quantifies the degree to which a sentence is personified. This confirms that indeed, our proposed **PINEAPPLE** method is successful in training a model to personify text.

Our PA-BART model also performs well for both BLEU and BERTScore, scoring better than the COMET and BART baselines, and coming second only to the human-written personifications.

Lastly, with regards to fluency, the BL-BART model outperforms the PA-BART model. This is likely because when considering GPT-2 likelihood, it may unfavorably penalize creative sentences with personifications since these are naturally less common in regular text. As an example, the sentence *"The stars danced playfully"* (GPT-2 loss = 7.02) would be deemed significantly less fluent than the sentence *"The stars twinkled brightly"* (GPT-2 loss = 5.24), even though they are both valid sentences with similar meanings. This argument is further supported by the fact that even the reference human-generated personifications received a lower fluency score than the BL-BART outputs. Further, literal sentences are indeed typically more *fluent* overall than personifications since they express the meaning literally. Nevertheless, we are still interested in the other qualities being measured by fluency: *Is the sentence coherent? Does it make unnecessary grammatical errors?* In this regard, the fluency of PA-BART remains quite good. It is significantly better than the fluency of the COMET personifications and only slightly worse than the fluency of the human-written personifications.

4.2 Human Evaluation Results

Human evaluation results are reported in Table 3. Out of the five human evaluation metrics, the most pertinent metric to the personification generation task is *Personificationhood*, as this metric explicitly tries to quantify the presence and overall quality of personifications. In this metric, our PA-BART model performs significantly better than both baselines and is only slightly worse than the human reference personifications. This indicates that PA-BART is very successful in generating personifications that humans are able to detect and understand.

Aside from measuring the presence of personifications, we also want to measure more fine-grained qualities of these personifications. This is done by considering the *Appropriateness* and *Interestingness* scores. In *Interestingness*, PA-BART significantly outperforms both baselines but is worse than human annotations, while in *Appropriateness*, PA-BART slightly outperforms BL-BART and is slightly worse than human annotations. Overall, we can conclude that the personifications generated by PA-BART are of good quallity: the ATTRIBUTES match up well with the TOPICS, and they are overall very creative. This is further exemplified through the qualitative examples explored in §4.3.

Observations from *Meaning Preservation* and *Fluency* are very similar to those from the BLEU/BERTScore/Fluency metrics in the automatic evaluations. For *Meaning Preservation*, PA-BART performs best among all models, and only slightly trails human references. Meanwhile, for fluency, BL-BART was ranked the most fluent, outperforming both PA-BART and the human references. As discussed previously, this is likely due to the fact that literal sentences are generally perceived to be more fluent than personifications.

4.3 Qualitative Analysis

Table 4 contains a list of color-coded qualitative examples for each method. In Figure 2, we previously outlined three main types of personification TOPIC-ATTRIBUTE pairs, namely the cases where ATTRIBUTE is a noun, a verb, and an adjective. The first three examples in Table 4 demonstrate the capacity of our PA-BART model to capture all three cases. In the first example, the literal verb in "*your phone **rings** out loud*" is replaced with the more appropriate animate verb in "*your phone **yells** out loud*." In the second, "*silence is **key***" is replaced with a noun in "*silence is **a ghost***", while in the third example, the literal adjective "*very **difficult***" is replaced with the animate adjective "*very **lonely***". These examples illustrate the generative flexibility of our model and its capacity to generate diverse outputs with different parts-of-speech.

	BLEU		BERTScore			
	Input	Gold	Input	Gold	Fluency ↓	Animacy
Human Annotation	0.172	1.000	0.749	1.000	5.264	0.332
COMET	0.127	0.128	0.655	0.569	6.366	-2.028
BL-BART	0.132	0.133	0.728	0.617	**4.573**	0.106
PA-BART	**0.153**	**0.160**	**0.748**	**0.636**	5.460	**0.679**

Table 2: Average automatic evaluation results. The best-scoring method for each metric is highlighted in **bold**. Higher scores are better for all metrics except for fluency.

	Personificationhood	Appropriateness	Fluency	Interestingness	Meaning Preservation
Human Annotation	3.763	4.175	4.138	3.667	3.913
COMET	3.525	3.563	3.738	1.801	3.550
BL-BART	3.500	3.938	**4.188**	2.006	3.750
PA-BART	**3.738**	**4.000**	4.138	**2.782**	**3.875**

Table 3: Average human evaluation results. The best-scoring method for each metric is highlighted in **bold**.

We also observe that the outputs for PA-BART generally capture the meaning of the original text (and surrounding context) more accurately than the other baselines. In fact, the personifications greatly enhance the expressiveness of some of these sentences. In the first example, PA-BART replaces '*rings*' with '*yells*', while COMET replaces it with '*beeps*', and BL-BART leaves '*rings*' unchanged and just adds more details. Given the context of the sentence, we see that '*yells*' is more appropriate, expressive, and consistent with the context. A similar argument can be made for most of the other examples in the table: for the third example, PA-BART replaces the literal *"very **difficult**"* with the much more animate and expressive *"very **lonely**"*, which is a suitable word to describe a relationship. In the fourth example, the BL-BART model is able to successfully capture the meaning of *"the house **became silent**"* with *"the house **fell into disrepair**"*. Although the meaning is correct, *"fell into disrepair"* is more literal and does not contain a personification. Compare this with the PA-BART's choice to replace *"the house **became silent**"* with *"the house **lamented**"*, which fits with the overall context (*"Then there were no more parties..."*), and also greatly enhances creativity by invoking the vivid image of lamentation. Meanwhile, in the fifth example, BL-BART personifies *"the crickets were **silent**"* with *"the crickets were **calling**"*. However, this shift completely changes the meaning, so it is a rather poor choice of personification. In contrast, PA-BART rewrites *"the air was **still**"* as *"the air was **tired**"*, which is a reasonable personification that is consistent with the imagery in the sentence (*"moonless nights"*, *"crickets were silent"*). Hence, we see that PA-BART can generate creative and meaningful personifications, while simultaneously staying true to the spirit of the sentence.

We also point out that our model is not limited to single-word substitutions. Rather, it considers a holistic view of the entire sentence and modifies key segments as necessary. This allows PA-BART to handle compound phrases well: consider, for instance, the one-to-many-word substitution of *'key'* $\longrightarrow$ *'a ghost'* (example 2), and the many-to-one-word substitution of *"became silent"* $\longrightarrow$ *"lamented"* (example 4). More importantly, PA-BART is also able to simultaneously generate personifications in two disjoint parts of the sentence, as seen in the last example: "*The sound **clapped** loud enough to make your ear **cry***." Here, there are two personifications in *"sound **hit**"* $\longrightarrow$ *"sound **clapped**"*, and *"ear **hurt**"* $\longrightarrow$ *"ear **cry**"*.

This last example also demonstrates the imperfection of our method. Although the model is able to generate two personifications, it loses a component of the original sentence because the recipient of the action (*'Frank'*) has disappeared. This same issue of meaning or information loss is present in example 2, where our model's output of *"silence is **a ghost**"*, while a personification, actually contradicts the original text *"silence is **key**"*. BL-BART's output of *"silence is **preferred**"*, while not a personification, correctly preserves the original meaning, as does the human reference of *"silence is **the protagonist**"*. This suggests that the model may still need some improvements with balancing creativity and semantic preservation. Other possible weaknesses are outlined in §6.

4.3.1 Novelty and Diversity Analysis

We randomly sample 30 examples from the PA-BART generations- and manually identify the parts

Method	Text
Literal Input	You are at a business dinner with your boss when your phone rings out loud (ex.1)
Human Ref	You are at a business dinner with your boss when your phone starts singing out loud
COMET	You are at a business dinner with your boss when your phone beep out loud
BL-BART	You are at a business dinner with your boss when your phone rings and you answer out loud
PA-BART	You are at a business dinner with your boss when your phone yells out loud
Literal Input	In most horror settings, silence is key. (ex.2)
Human Ref	In most horror settings, silence is the protagonist.
COMET	In most horror settings, silence scary.
BL-BART	In most horror settings, silence is preferred.
PA-BART	In most horror settings, silence is a ghost.
Literal Input	Her relationships with family and friends are very difficult (ex.3)
Human Ref	Her relationships with family and friends behave very unusually
COMET	Her relationships with family and friends serious
BL-BART	Her relationships with family and friends have always been strong.
PA-BART	Her relationships with family and friends are very lonely
Literal Input	Then there weren't any more parties as the house became silent (ex.4)
Human Ref	Then there weren't any more parties as the house kept mum.
COMET	Then there weren't any more parties as the house build.
BL-BART	Then there weren't any more parties as the house fell into disrepair.
PA-BART	Then there were no more parties as the house lamented.
Literal Input	It was a moonless nights, the air was still and the crickets were silent (ex.5)
Human Ref	It was a moonless nights, the air was asleep and the crickets were silent
COMET	It cold outside a moonless nights, the air cold outside still and the crickets noisy
BL-BART	It was a moonless nights, the air was still and the crickets were calling.
PA-BART	It was one of those moonless nights, the air was tired and the crickets were silent
Literal Input	The sound hit Frank loud enough to make your ear hurt (ex.6)
Human Ref	The sound slapped Frank loud enough to make your ear hurt
COMET	The sound echo Frank loud enough to make your ear sense sound
BL-BART	The sound of Frank Sinatra is loud enough to make your ear ring.
PA-BART	The sound clapped loud enough to make your ear cry

Table 4: Qualitative examples for personification: literal input, human writing, COMET, BL-BART, and PA-BART. More can be found in Appendix D.

of the sentences that were personified, as well as the animate ATTRIBUTES used to personify the TOPICS. Among the 30 examples, there were 27 unique ATTRIBUTES, and only 3 repeats. Additionally, there were 9 examples which generated completely new ATTRIBUTES that were never before seen in the training set, which demonstrates that the model is able to sufficiently capture the essence of a personification, rather than just blindly memorizing ATTRIBUTES from the training data.

5 Related Work

We present the linguistic underpinnings behind the TOPIC-ATTRIBUTE framework used in this paper and explore how other types of figures of speech are generated. We also explore what makes personification generation so challenging.

Linguistic Motivations. Personifications traditionally do not have clearly defined classifications. In fact, even within the linguistic community, the definition of a personification is not always very clear-cut (Edgecombe, 1997; Hamilton, 2002). A study by Long (2018) examines the personification structure "*nonhuman subject + predicate verb (used for humans only) + others*," as well as the structure "*others + predicate verb (used for humans only) + nonhuman object + others*." We generalize and repackage these concepts, renaming the *subject* as the TOPIC and the *predicate verb* as the ATTRIBUTE. In doing so, we are able to capture more general notions of animacy beyond just verbs.

Generation of Metaphors, Similes, etc. A lot of studies on metaphors have focused on identification using techniques like word sense disambiguation (Birke and Sarkar, 2007), topic modeling (Strzalkowski et al., 2013; Heintz et al., 2013), dependency structures (Jang et al., 2015), and semantic analysis (Hovy et al., 2013). In terms of generation, early systems have explored grammar rules (Gargett and Barnden, 2013), while more recently, large language models have greatly aided in metaphor generation. Most notably, Stowe et al. (2021) generate metaphors by considering conceptual mappings between certain domains and verbs. Chakrabarty et al. (2021) further build on this by creating a parallel corpus of metaphors and training a large language model to perform the generation.

We also note here that the two aforementioned studies already cover personifications to a certain extent. However, these studies considered personifications as subtypes of metaphors. Some of the methods used may not generalize well to other types of personifications. Our study is the first to focus specifically on generating personifications.

For generating similes, Chakrabarty et al. (2020) propose using style-transfer models with COMET commonsense knowledge to generate similes. The study similarly creates a parallel corpus and trains a seq2seq model to perform the generation.

There is also a recent work by Keh et al. (2022) that uniquely investigates the generation of tongue twisters using seq2seq and language models.

Personifications. There are currently few studies that specifically work on personifications. Gao et al. (2018) detect personifications as a subtype of metaphors, but not as its own figure of speech. Generation is largely unexplored. We believe this is likely because personifications are generally more difficult to define and categorize. Furthermore, because several sources simplify personifications to fall under metaphors (Stowe et al., 2021; Chakrabarty et al., 2021), there is also a lack of personification-specific datasets.

Constrained Text Generation. There is also a body of work exploring the family of more gen-

eral constrained text generation tasks. Gangal et al. (2022) investigate NAREOR, or narrative ordering, which rewrites stories in distinct narrative orders while preserving the underlying plot. Miao et al. (2019) show gains on several tasks through determining Levenshtein edits per generation step using Metropolis-Hastings sampling. Feng et al. (2019) propose Semantic Text Exchange to modify the topic-level semantics of a piece of text.

Lin et al. (2020) propose CommonGen, a generative commonsense reasoning task based on concept-to-text generation. Works investigating this task include EKI-BART (Fan et al., 2020) and KG-BART (Liu et al., 2021), which use external knowledge to enhance performance on CommonGen. SAPPHIRE (Feng et al., 2021b) uses the data itself and the model's own generations to improve CommonGen performance, while VisCTG (Feng et al., 2022) uses per-example visual grounding.

6 Conclusion and Future Work

In this paper, we explored the task of personification generation. We curated a dataset of personifications and proposed the **PINEAPPLE** method to automatically de-personify text. Using our parallel corpus, *PersonifCorp*, we trained a seq2seq model (BART) to generate creative personifications. Through automatic, human, and qualitative evaluation, we demonstrated that these personifications make sentences more interesting and enhance the text's overall appeal. Our finetuned model successfully does this while maintaining a high level of fluency and meaning perservation.

Some weaknesses of our model include failing to personify more complex sentence structures, and occasionally failing to preserve the exact meaning of the original sentence. We also believe that our model still has room to grow in terms of the diversity of personifications generated. Further, we can explore unsupervised style transfer methods (Yang et al., 2018; Malmi et al., 2020; Krishna et al., 2020), where we regard the personification-hood of a sentence as a kind of style. We can also investigate data augmentation methods (Feng et al., 2021a, 2020; Dhole et al., 2021) to further expand our dataset. Another promising direction would be to explore ways to acquire more control over which parts of the sentence are personified or what types of personifications are generated, or to apply this to make dialogue agents more interesting, e.g. by giving them more personality (Li et al., 2020).

Ethics Statement

Our human and automatic evaluations (see §3.2) are done over content either directly sourced from, or generated by publically available, off-the-shelf pretrained models trained either on already existing, publicly available datasets, or datasets further derived by post-processing the same — as further described in Datasets (see §2 for more).

We do collect human evaluation ratings using crowd-sourcing, specifically through AMT and in-person annotation. However, we neither solicit, record, nor request any kind of personal or identity information from the annotators. Our AMT annotation was conducted in a manner consistent with terms of use of any sources and intellectual property and privacy rights of AMT crowd workers. Crowdworkers were fairly compensated: $1.12 per fluency + appropriateness + meaning preservation evaluation HIT, and $0.56 per personificationhood evaluation HIT, for roughly 6 min (first) and 2 min (latter) tasks, respectively. This is at least 1.5-2 times the minimum U.S.A. wage of $7.25 per hour ($0.725 per 6 minutes and $0.25 per 2 minutes).

We primarily perform experiments on personification in English (Bender and Friedman, 2018).

NLG models are known to suffer from biases learnable from training or finetuning on data, such as gender bias (Dinan et al., 2020). However, our work and contribution does not present or release any completely new model architechtures, and is primarily concerned with more careful adaptation and finetuning of existing pretrained models for a particular class of figurative construct (i.e. personification). The frailties, vulnerabilities, and potential dangers of these models have been well researched and documented, and a specific re-investigation would be repetitive and beyond the scope and space constraints of this paper.

We do not foresee any explicit way that malicious actors could specifically misuse fintuned models that could be trained on our data, beyond the well-researched, aforementioned misuse that is possible in general with their instantiation for any transduction task or dataset (e.g. summarization).

References

Emily M Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Julia Birke and Anoop Sarkar. 2007. Active learning for the identification of nonliteral language. In *Proceedings of the Workshop on Computational Approaches to Figurative Language*, pages 21–28, Rochester, New York. Association for Computational Linguistics.

Morton W Bloomfield. 1980. Personification-metaphors. *The Chaucer Review*, pages 287–297.

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.

Tuhin Chakrabarty, Smaranda Muresan, and Nanyun Peng. 2020. Generating similes effortlessly like a pro: A style transfer approach for simile generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6455–6469, Online. Association for Computational Linguistics.

Tuhin Chakrabarty, Xurui Zhang, Smaranda Muresan, and Nanyun Peng. 2021. MERMAID: Metaphor generation with symbolism and discriminative decoding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4250–4261, Online. Association for Computational Linguistics.

Jianfu Chen, Polina Kuznetsova, David Warren, and Yejin Choi. 2015. Déjà image-captions: A corpus of expressive descriptions in repetition. In *HLT-NAACL*, pages 504–514.

Kaustubh D Dhole, Varun Gangal, Sebastian Gehrmann, Aadesh Gupta, Zhenhao Li, Saad Mahamood, Abinaya Mahendiran, Simon Mille, Ashish Srivastava, Samson Tan, et al. 2021. Nl-augmenter: A framework for task-sensitive natural language augmentation. *arXiv preprint arXiv:2112.02721*.

Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston. 2020. Queens are powerful too: Mitigating gender bias in dialogue generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8173–8188.

Aletta G Dorst. 2011. Personification in discourse: Linguistic forms, conceptual structures and communicative functions. *Language and Literature*, 20(2):113–135.

Rodney Stenning Edgecombe. 1997. Ways of personifying. *Style*, 31(1):1–13.

Zhihao Fan, Yeyun Gong, Zhongyu Wei, Siyuan Wang, Yameng Huang, Jian Jiao, Xuanjing Huang, Nan

Duan, and Ruofei Zhang. 2020. An enhanced knowledge injection model for commonsense generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2014–2025, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Steven Y. Feng, Varun Gangal, Dongyeop Kang, Teruko Mitamura, and Eduard Hovy. 2020. GenAug: Data augmentation for finetuning text generators. In *Proceedings of Deep Learning Inside Out (DeeLIO): The First Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 29–42, Online. Association for Computational Linguistics.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021a. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.

Steven Y. Feng, Jessica Huynh, Chaitanya Prasad Narisetty, Eduard Hovy, and Varun Gangal. 2021b. SAPPHIRE: Approaches for enhanced concept-to-text generation. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 212–225, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Steven Y Feng, Aaron W Li, and Jesse Hoey. 2019. Keep calm and switch on! preserving sentiment and fluency in semantic text exchange. 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 2701–2711.

Steven Y. Feng, Kevin Lu, Zhuofu Tao, Malihe Alikhani, Teruko Mitamura, Eduard Hovy, and Varun Gangal. 2022. Retrieve, caption, generate: Visual grounding for enhancing commonsense in text generation models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10618–10626.

Mary C Flannery. 2016. Personification and embodied emotional practice in middle english literature. *Literature Compass*, 13(6):351–361.

Varun Gangal, Steven Y. Feng, Malihe Alikhani, Teruko Mitamura, and Eduard Hovy. 2022. Nareor: The narrative reordering problem. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10645–10653.

Ge Gao, Eunsol Choi, Yejin Choi, and Luke Zettlemoyer. 2018. Neural metaphor detection in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 607–613, Brussels, Belgium. Association for Computational Linguistics.

Andrew Gargett and John Barnden. 2013. Gen-meta: Generating metaphors using a combination of ai reasoning and corpus-based modeling of formulaic expressions. pages 103–108.

Craig A. Hamilton. 2002. Mapping the mind and the body: On w. h. auden's personifications. *Style*, 36(3):408–427.

Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. CASCADE: Contextual sarcasm detection in online discussion forums. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1837–1848, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ilana Heintz, Ryan Gabbard, Mahesh Srivastava, Dave Barner, Donald Black, Majorie Friedman, and Ralph Weischedel. 2013. Automatic extraction of linguistic metaphors with LDA topic modeling. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 58–66, Atlanta, Georgia. Association for Computational Linguistics.

Dirk Hovy, Shashank Srivastava, Sujay Kumar Jauhar, Mrinmaya Sachan, Kartik Goyal, Huying Li, Whitney Sanders, and Eduard Hovy. 2013. Identifying metaphorical word use with tree kernels. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 52–57, Atlanta, Georgia. Association for Computational Linguistics.

Nikhil Jaiswal. 2020. Neural sarcasm detection using conversation context. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 77–82, Online. Association for Computational Linguistics.

Hyeju Jang, Seungwhan Moon, Yohan Jo, and Carolyn Rosé. 2015. Metaphor detection in discourse. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 384–392, Prague, Czech Republic. Association for Computational Linguistics.

Sedrick Scott Keh, Steven Y. Feng, Varun Gangal, Malihe Alikhani, and Eduard Hovy. 2022. Pancetta: Phoneme aware neural completion to elicit tongue twisters automatically. *arXiv preprint*.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Empirical Methods in Natural Language Processing*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Aaron W. Li, Veronica Jiang, Steven Y. Feng, Julia Sprague, Wei Zhou, and Jesse Hoey. 2020. Aloha: Artificial learning of human attributes for dialogue agents. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8155–8163.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, Online. Association for Computational Linguistics.

Ye Liu, Yao Wan, Lifang He, Hao Peng, and Philip S. Yu. 2021. Kg-bart: Knowledge graph-augmented bart for generative commonsense reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(7):6418–6425.

Deyin Long. 2018. Meaning construction of personification in discourse based on conceptual integration theory. *Studies in Literature and Language*, 17:21–28.

Eric Malmi, Aliaksei Severyn, and Sascha Rothe. 2020. Unsupervised text style transfer with padded masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8671–8680, Online. Association for Computational Linguistics.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. 2019. Cgmh: Constrained sentence generation by metropolis-hastings sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6834–6842.

Vlad Niculae and Cristian Danescu-Niculescu-Mizil. 2014. Brighter than gold: Figurative language in user generated comparisons. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2008–2018, Doha, Qatar. Association for Computational Linguistics.

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 of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI*.

Kevin Stowe, Tuhin Chakrabarty, Nanyun Peng, Smaranda Muresan, and Iryna Gurevych. 2021. Metaphor generation with conceptual mappings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6724–6736, Online. Association for Computational Linguistics.

Tomek Strzalkowski, George Aaron Broadwell, Sarah Taylor, Laurie Feldman, Samira Shaikh, Ting Liu, Boris Yamrom, Kit Cho, Umit Boz, Ignacio Cases, and Kyle Elliot. 2013. Robust extraction of metaphor from novel data. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 67–76, Atlanta, Georgia. Association for Computational Linguistics.

Enrica Troiano, Carlo Strapparava, Gözde Özbal, and Serra Sinem Tekiroğlu. 2018. A computational exploration of exaggeration. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3296–3304, Brussels, Belgium. Association for Computational Linguistics.

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.

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P. Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 7298–7309, Red Hook, NY, USA. Curran Associates Inc.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix A: De-Personification Pipeline

A.1 Dependency Tree Merging Example

Figure 4 contains an example of the merging process that was described in the TOPIC-ATTRIBUTE extraction step in §2. As outlined in §2, edge relations to iteratively merge are *negation modifiers*, *possession modifiers*, *nominal modifiers*, *adjectival complements*, and *objects of prepositions*, as well as words tagged as determiners and parts of compound phrases. The priority order for merging is as follows: 1) compound phrases, 2) nominal modifiers, 3) possession modifiers, 4) negation modifiers, 5) determiners, 6) objects of prepositions, 7) adjectival complements.

A.2 Human-Related Words

In §2, we defined the *IsAPerson* animacy metric as the average of the *IsA* scores between the TOPIC and various words that are very closely related to *'human'*. We called this set the HUMANSET. The words contained in HUMANSET are as follows: {"person", "human", "man", "woman", "human being", "boy", "girl"}.

These words were empirically selected by considering a list of synonyms of the word *'person'* and checking the *IsA* relation COMET scores with the word *'human'*. All of the above words have *IsA* scores with *'person'* of less than 5.10.

A.3 Parameters and Thresholds

***IsAPerson* Threshold.** For the *IsAPerson* animacy score, we use a threshold of 7.0. *IsAPerson* scores < 7.0 are considered animate, while scores ≥ 7.0 are considered inanimate. This threshold was selected empirically using words known to be animate and words known to be inanimate. Words tested include "she" (5.31), "person" (6.41), "moon" (8.743), "opportunity" (9.488), "stars" (8.717), "joe" (5.804), "jane" (4.976), "the police officer" (6.462), "my friend" (6.805), "my new iphone" (10.055). From these observations, we observe that all animate words have an *IsAPerson* score of < 7.0, while all inanimate objects have a score of ≥ 7.0. We hence conclude that 7.0 is a suitable threshold.

Candidate Selection Composite Score Parameters. For the α, β, γ used in the composite score for candidate selection, we use values of $\alpha = 1, \beta = 1, \gamma = 1$. This was selected for two reasons. First, all of the score values had largely similar scales (logarithmic), so setting α, β, γ to a larger value like 2 or 3 would disproportionately favor a certain metric, which is not what we desire. Second, we experimented with using values such as 0.8, 1.2, and 1.5, but the generated de-personifications were either very similar or slightly worse than the default setting of $\alpha = 1, \beta = 1, \gamma = 1$. A possible future direction would be to explore possible values of α, β, γ more thoroughly, but for this dataset, we stick to the simple case of $\alpha = 1, \beta = 1, \gamma = 1$.

Metric	Spearman Correlation	Krippendorff α
Personificationhood	0.0934	0.0250
Appropriateness	0.1660	0.1778
Fluency	0.0050	0.0942
Interestingness	0.6160	0.5898
Meaning Preservation	0.0389	0.2558

Table 5: Inter-annotator agreement scores.

B Appendix C: Evaluation Details

B.1 Human Evaluation Setup

A total of 20 unique AMT annotators participated in the study for fluency, appropriateness, and meaning preservation, each performing 4.0 HITs on average. Annotators were compensated 1.12$ per HIT, each of which was designed to take <6 mins on average.

22 unique AMT annotators participated in the second, separate study for personificationhood, each performing 4.36 HITs on average. Annotators were compensated 0.56$ per HIT, each of which was designed to take <2 mins on average.

For the interestingness study, the details regarding annotator background and selection can be found in §3.2.2 and Appendix B.3.

The html templates including instructions, questions and other study details corresponding to both these AMT studies can be found in the `templates/` subfolder of our code submission zip, with the names `fluency_appropriateness_meaningPreservation.html` and `personificationhood.html` respectively.

B.2 Inter-Annotator Agreement Scores

Each generated input instance and its respective model outputs are labelled by two distinct annotators. To measure inter-annotator agreement, we use Spearman correlation and Krippendorff α, as reported in Table 5.

To get the Spearman correlation point value for a given aspect and test instance, we compute mean pairwise Spearman correlation between the aspect values assigned to the corresponding model outputs by every pair of annotators. Specifically, we use the *scipy.stats* implementation to compute this. [10]

For Krippendorff α, we treat each human evaluation aspect as an ordinal quantity. Specifically, we use the implementation provided by the python library *krippendorff 0.5.1*.[11]

B.3 English Assessment Test for Annotators

From the native English-speaking university student annotators who enrolled to participate in our Interestingness study, we first elicited answers to an English assessment test, as mentioned in §3.2.2.

The assessment test comprised of 12 questions spanning multiple question types testing the examinee's grasp of the use and distinction between various figures of speech, basic literary general knowledge, and verbal reasoning skills. A spreadsheet file containing this test can be found with the name *LiteratureTest.xlsx* under the *Templates/* subfolder of our code submission .zip file.

The final annotators used for our interestingness study were chosen from those who got 11 or more of the 12 questions on the English assessment test correct, hence scoring at least 85% on the test.

C Appendix B: Implementation Details

The BART-base model was trained using a learning rate of 2e-5. This was by conducting a hyperparameter search over the values {1e-6, 5e-6, 1e-5, 2e-5, 5e-5, 1e-4} and selecting the model/epoch based on lowest validation loss. The same process was done to select a batch size of 4 using a hyperparameter search over values {2,4,8,16}. Training was done for 20 epochs and 400 warmup steps. The Adam optimizer was used, and inputs were truncated to a maximum length of 64 tokens (using BART's subword tokenization).

Training was done on Google Colaboratory environments using V100 GPUs. For the BART-base model, a single training loop of 20 epochs takes approximately 10 minutes to complete.

[10] https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.spearmanr.html

[11] https://pypi.org/project/krippendorff/

Method	Text
Literal Input	The news hit me hard. (ex.7)
Human Ref	The news punched me hard.
COMET	The news report event late me hard.
BL-BART	The news hit me hard.
PA-BART	The news pounded me hard.
Literal Input	Panic is sweeping through the streets contagiously. (ex.8)
Human Ref	Panic is driving around the streets contagiously.
COMET	Panic is result through the streets contagiously.
BL-BART	Panic is running through the streets contagiously.
PA-BART	Panic runs through the streets contagiously.
Literal Input	In the dead of summer, in Ohio, the clothes felt hot and humid. (ex.9)
Human Ref	In the dead of summer, in Ohio, the clothes seemed to be sweating themselves.
COMET	In the dead of summer, in Ohio, the clothes fit and humid.
BL-BART	In the dead of summer, in Ohio, the clothes we wear can be hot and humid.
PA-BART	In the dead of summer, in Ohio, the clothes complained and humid.
Literal Input	Streaks of electricity zipped through the air, swirling around and around dangerous (ex.10)
Human Ref	Streaks of electricity cackled through the air, swirling around and around dangerous.
COMET	Streaks of electricity stain carpet through the air, swirling around and around dangerous.
BL-BART	Streaks of electricity were shooting through the air, swirling around and around the building.
PA-BART	Streaks of electricity danced around and around dangerous.
Literal Input	The sun is nowhere to be found, and the streets are eerily quiet (ex.11)
Human Ref	The sun seems to be on leave, and the streets are eerily quiet.
COMET	The sun down, and the streets full of traffic.
BL-BART	The sun, and the streets.
PA-BART	The sun refuses to be found, and the streets are polite
Literal Input	The wind blew through me fast. (ex.12)
Human Ref	The wind stampeded through me fast.
COMET	The wind howl through me fast.
BL-BART	The wind was going through me fast.
PA-BART	The wind ran me fast.

Table 6: Additional qualitative examples for personification outputs: literal input, human writing, COMET, BL-BART, and PA-BART.

D Appendix D: Additional Examples

Table 6 is an extension of Table 4 and contains additional qualitative examples of the generations.

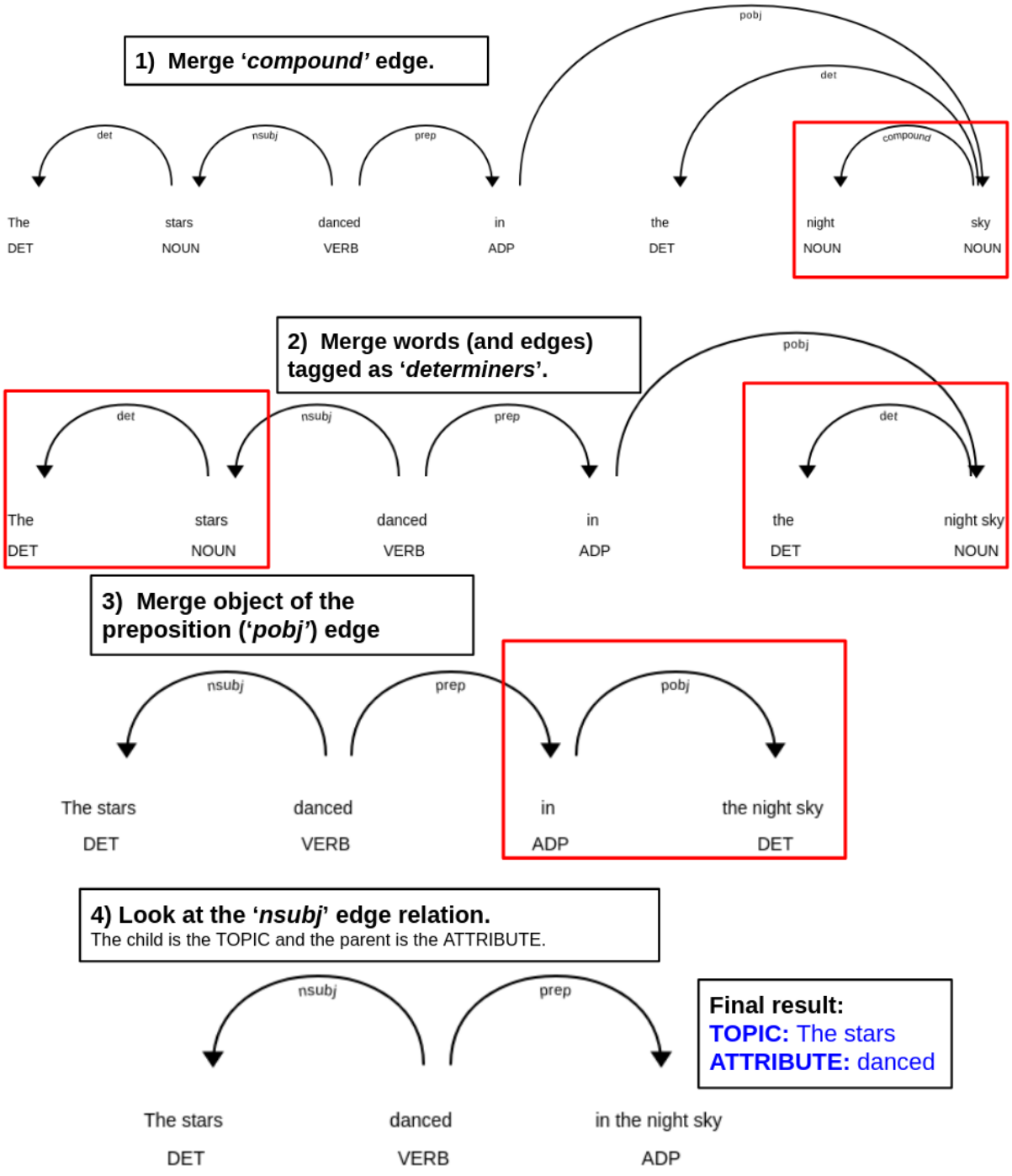

Figure 4: Step-by-step example of the merging process for TOPIC-ATTRIBUTE identification.