Harnessing Privileged Information for Hyperbole Detection

Rhys Biddle[♠♣] **Maciej Rybiński**[♣] **Qian Li**[♠] **Cécile Paris**[♣] **Guandong Xu**[♠]
[♠]Advanced Analytics Institute, University of Technology Sydney, Australia
[♣]CSIRO Data61, Sydney, Australia
rhys.biddle@student.uts.edu.au
{firstname.lastname}@uts.edu.au
{firstname.lastname}@csiro.au

Abstract

The detection of hyperbole is an important stepping stone to understanding the intentions of a hyperbolic utterance. We propose a model that combines pre-trained language models with privileged information for the task of hyperbole detection. We also introduce a suite of behavioural tests to probe the capabilities of hyperbole detection models across a range of hyperbole types. Our experiments show that our model improves upon baseline models on an existing hyperbole detection dataset. Probing experiments combined with analysis using local linear approximations (LIME) show that our model excels at detecting one particular type of hyperbole. Further, we discover that our experiments highlight annotation artifacts introduced through the process of literal paraphrasing of hyperbole. These annotation artifacts are likely to be a roadblock to further improvements in hyperbole detection.

1 Introduction

The analysis of figurative language by Natural Language Processing (NLP) systems is a challenge confronting researchers and practitioners (Reyes and Rosso, 2014; Rai and Chakraverty, 2020). Hyperbole is a common type of figurative language that is defined by an intentionally excessive contrast between utterance meaning and reality along a semantic scale to convey an evaluation (e.g., '*my bedroom is the size of a postage stamp*') (McCarthy and Carter, 2004; Mora, 2009; Claridge, 2010; Carston and Wearing, 2015; Burgers et al., 2016). The detection of hyperbole has proven to be a challenging problem for NLP systems, much like the detection of other figures of speech (Troiano et al., 2018; Kong et al., 2020; Abulaish et al., 2020). The evaluative nature of hyperbole motivates the importance of understanding hyperbole for affective computing applications (e.g., sentiment analysis).

Learning under Privileged Information (LUPI) is a learning paradigm that involves providing additional information during training to help teach a model to learn a particular phenomenon (Pechyony and Vapnik, 2010). The source and type of privileged information (PI) varies depending on application, such as a list of ingredients present in an image to help teach a computer vision model to detect food in images (Meng et al., 2019), or the human ratings of various aesthetic categories of images for automated assessment of aesthetic photo quality (Shu et al., 2020). We propose to use literal paraphrases of hyperbole as a source of PI for hyperbole detection. We hypothesise that this information will help a model to learn the excessive contrast within a particular hyperbole (e.g., '*my head is **exploding** right now*' → '*my head is **hurting** right now*').

Our contributions in this paper are as follows; (1) We propose a method for hyperbole detection based on the injection of PI; (2) We introduce **HyperProbe**, a suite of behavioural tests for hyperbole detection models; (3) We reveal that annotation artifacts are a potential roadblock for progress on hyperbole detection.

2 HYPO

The **HYPO** dataset is an annotated collection of hyperbole introduced by Troiano et al. (2018). The dataset consists of manually composed hyperbole and hyperbole sourced from various online sources including click-bait headlines, love letters, advertisements, and animated cartoons.

Annotation for **HYPO** was carried out by crowd workers who were given several tasks based on each example. The crowd workers had to assess whether they thought the utterance contained hyperbolic content. A follow up task was to highlight the specific words in the utterance they considered

Hyperbole Corpus	Paraphrase Corpus	Minimal Units Corpus
The principal is unhappy...we're **cooked**.	The principal is unhappy...we're *in trouble*.	Well **cooked** vegetables can be pureed easily.
Her morning jog turned into a **marathon**	Her morning jog turned into a *long run*	There was a **marathon** in the city today

Table 1: **HYPO examples**. **Hyperbole Corpus** contains original hyperbolic utterances. **Paraphrase Corpus** contains a literal paraphrase. **Minimal Units Corpus** contains examples that contain the hyperbolic words/phrases in a non-hyperbolic context.

Type	Keywords
ECF	absolute, complete, entire, pure, whole, impossible, never, no, nobody, nowhere, perfect, flawless, endless, eternal, infinite, all, always, every, everybody, everyone, everywhere, definite, exact, undeniable
Quantitative	small, big, slow, fast, thin, thick, tall, length, large, high
Qualitative	bad, corrupt, evil, fraud, wicked, chaos, confusion, disorder, garbage, riot, dead, hell, misery, murder, nightmare, alarm, fear, panic, scared, shock, anxiety, autism, blind, deaf, insomnia, bitter, pierce, sharp, spicy, toxic, cancer, fever, headache, pain, sad, suffer, attack, explode, fight, rape, ruin, wreck, dream, heaven, paradise, utopia, vital, attract, beauty, charm, grace, handsome, amaze, good, great, ideal, impress

Table 2: **Hyperbole term lists**. **Type** refers to the type of hyperbole as defined by Mora (Mora, 2009). **Keywords** is a list of the keywords in word list.

to be hyperbolic. Additionally, the workers were then asked to paraphrase the original hyperbolic sentence such that it was no longer hyperbolic.

The worker responses to the first task were used to filter out non-hyperbolic utterances resulting in 709 hyperbolic utterances in total, denoted as the Hyperbole Corpus. The list of hyperbolic tokens identified by the crowd workers was used to create a second corpus, denoted the Minimal Units Corpus (709 sentences). The literal paraphrases also made up another corpus, the Paraphrase Corpus (709 sentences). Combining these three corpora, every hyperbolic utterance in the Hyperbole Corpus has two non-hyperbolic counterparts from the Minimal Units Corpus and Paraphrase Corpus respectively, see Table 1. In total just over 2.1k sentences make up the final version of **HYPO**.

3 HyperProbe

Our **HyperProbe** suite consists of synthetic data generated to probe the ability of models to detect hyperbole[1]. The suite is created to target the three types of hyperbole identified by (Mora, 2009): **Extreme Case Formulations** (ECF), **Qualitative Hyperbole** and **Quantitative Hyperbole**. The creation of test sentences follows a general four step procedure:

1. **Word List Creation**: we create seed word lists containing words to be used in test sentences. These seed word lists are divided by part-of-speech class and are created based on the word lists curated by Mora (2009), see Table 2, for hyperbole-prone words.

2. **Sentence Template Creation**: we create syntactic templates to be filled by a sentence generator. The syntax for sentence templates is as follows; {TAG} indicates that a word is drawn from a user-defined seed word list (based on part-of-speech tags), **{TAG}** indicates that the word is drawn from a user-defined *hyperbole-prone* seed word list, {MASK} indicates that RoBERTa (Liu et al., 2019) will infill this token, a functionality provided by the CheckList framework (Ribeiro et al., 2020)[2].

3. **Test Sentence Generation**: consists of the generation of test sentences, via CheckList, using the word lists and templates generated in the previous steps.

4. **Manual Assessment and Annotation**: we assess the grammar and semantics of the generated test sentences and annotate the sentences. Our annotation consists of a binary label indicating the presence of hyperbolic content.

3.1 Extreme Case Formulation Tests

ECFs are semantic formulations that invoke extreme descriptions of events or objects (Whitehead, 2015; Pomerantz, 1986). A simple example of an ECF is a sentence that contains an extreme description via an adjective (*absolute*, *entire*, *infinite*, etc.), adverb (*always*, *never*, etc.), quantifier

[1] https://github.com/biddle-r/HyperProbe

[2] https://github.com/marcotcr/checklist

(*all*, *none*, etc.) or indefinite pronoun (*everybody*, *nobody*, etc.) (Edwards, 2000; Norrick, 2004). The intentionally non-literal use of ECFs has been identified as a rich source for hyperbolic expressions (McCarthy and Carter, 2004; Norrick, 2004; Mora, 2009; Whitehead, 2015; Carston and Wearing, 2015). The detection of ECFs is a fundamental requirement for a hyperbole detection model, and we design a set of test sentences to probe this ability. Given that ECF prone-words from Table 2 belong to various word classes and can appear in a myriad of grammatical patterns, we design several sentence templates, see Table 3. Upon completion of assessment and annotation there were 181 test sentences, 95 (52%) of which were labelled as hyperbolic, see Table 3.

3.2 Qualitative Hyperbole Tests

Qualitative hyperboles align with the subjective-emotional dimension of hyperbole (Mora, 2009). A subjective evaluation made to an excessive degree is the defining feature of qualitative hyperboles (e.g., '*this video is* ***cancer***', '*Sweet n sour chicken is* ***God Tier***'). The ability to detect and interpret qualitative hyperbole is a fundamental requirement of a hyperbole detection model. From the list of qualitative terms in Table 2, we compile a list containing 54 adjectives. We create six sentence templates to incorporate the adjectives into a sentence, see Table 4. Upon completion of assessment and annotation there were 306 test sentences, 87 (28%) of which were labelled as hyperbolic, see Table 4.

3.3 Quantitative Hyperbole Tests

Quantitative hyperboles align with the objective-gradational dimension of hyperbole (Mora, 2009). The defining feature of this type of hyperbole is the up-scaling of an *obvious* quantity or magnitude to an excessive degree (e.g., '*i have a* ***million*** *things left to do*', '*this year has felt like a* ***decade***'). We design a set of test sentences that allows us to probe the ability of models to detect hyperbolic expressions along quantitative dimensions. We use the list of quantitative terms in Table 2 and their comparative forms (e.g., *bigger*, *smaller*, *lighter*, etc.) as seed word lists for these sentences. We create two sentence templates to incorporate these into a sentence, see Table 5. Upon completion of assessment and annotation there were 43 test sentences, 21 (48%) of which were labelled as hyperbolic, see Table 5.

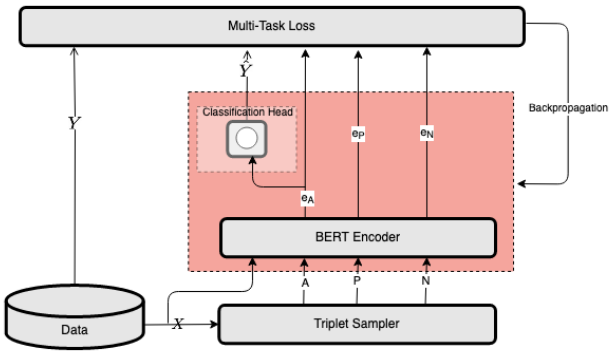

Figure 1: **BERT+PI**. Model contains a BERT encoder, a linear classification head and a Triplet Sampler. We incorporate PI via the triplet sampler.

4 Privileged Information for Hyperbole Detection

Our motivation for incorporating privileged information into a hyperbole detection model is based on observations from the foundational work of Troiano et al. (2018). The authors found that models trained on hyperboles and literal paraphrases performed marginally better on the task of hyperbole detection than models trained on hyperboles and non-literal sentences that used the hyperbolic words/phrases in a non-hyperbole context. We propose that treating literal paraphrases as privileged information and incorporating this information into a hyperbole detection model could improve the ability of a model to detect when a word or phrase was being used in an excessive hyperbolic manner.

In our proposed model, **BERT+PI**, we incorporate privileged information via triplet loss. We utilise a triplet loss because we want to force our model to differentiate between hyperbolic and non-hyperbolic usage of words and phrases, and we can strictly enforce this via triplet loss. Specifically, by specifying a hyperbolic sentence as an anchor sample, another hyperbole as a positive sample and a manually composed literal paraphrase (i.e., PI) as a negative sample, we are enforcing this difference in representation space.

4.1 BERT+PI

BERT+PI is based on a multi-task text classification framework. We use a triplet sampling module to sample negative and positive sentences for each sentence in the dataset. We use BERT (**?**) to encode a representation for each of these sentences and send the representation of the original sentence to a linear classification head. Representations of

Template	Example
{DT}{MASK}{MASK}{VB}{**JJ**}	the dishonest words are **endless**
{DT}{**JJ**}{MASK}{VB}{MASK}	the **endless** combinations are daunting
{DT}{MASK}{MASK}{**RB**}{VB_a}	the code was **never** cracked
{DT}{MASK}{MASK}{**RB**}{MASK}	the good times **always** roll
{DT}{MASK}{VB_l}{**RB**}{MASK}	the dog was **never** silent
{DT}{MASK}{MASK}{VB_l}{**RB**}	the drug problem is **everywhere**
{DT}{MASK}{MASK}{**DT**}{MASK}	The mother of **every** invention
{**DT**}{MASK}{MASK}{IN}{MASK}	**all** rights reserved in copyright
{**DT**}{MASK}{VB}{MASK}{MASK}	**every** child will be impacted
{DT}{MASK}{MASK}{MASK}{**PRON**}	The law applies to **everybody**
{**PRON**}{IN}{DT}{MASK}{VB}{MASK}	**nobody** on the street is home

Table 3: **Extreme Case Formulation Test Examples**. ***Template*** shows templates as provided to CheckList, ***Example*** is an example sentence as generated by CheckList.

Template	Example
{DT}{MASK}{MASK}{VB}{MASK}{**JJ**}	a world that is truly **wicked**
{DT}{MASK}{VB}{**JJ**}	The argument is **confusing**
{DT}{MASK}{VB}{MASK}{**JJ**}	The wine is very **bitter**
{DT}{MASK}{MASK}{VB}{**JJ**}	the oil residue is **toxic**
{DT}{**JJ**}{MASK}{VB}{MASK}	A **great** story was completed
{DT}{**JJ**}{MASK}{VB}{MASK}{MASK}	The **shocking** video was posted here

Table 4: **Qualitative Adjectives Test Examples**. ***Template*** shows templates as provided to CheckList, ***Example*** is an example sentence generated by CheckList.

Algorithm 1 Semi-Random Triplet Sampling

Require: $D = [t_0, t_1, ..., t_n]$
Require: $s \in \mathbb{Z}^+$ ▷ Sampling Factor
$H \leftarrow t \forall t \in D \mid t.\text{label} == 1$ ▷ t.label contains annotated label for t
$P \leftarrow t \forall t \in D \mid t.\text{label} == 2$ ▷ N consists of literal paraphrases (i.e., PI)
$S \leftarrow \emptyset$
for $i = 0, i < |D|, i++$ **do**
  $a \leftarrow D_i$
  $T \leftarrow \emptyset$
  for $j = 0, j < s, j++$ **do**
    if a.label $== 1$ **then**
      $p \leftarrow$ sample(H) ▷ sample(X) draws a random sample from X
      $n \leftarrow p$.par ▷ t.par is a literal paraphrase of t
    else if a.label $== 0$ **then**
      $p \leftarrow$ sample(P)
      $n \leftarrow p$.hyp ▷ t.hyp is a hyperbolic expression of t
    end if
    T.insert($[a, p, n]$)
  end for
  S.insert(T)
end for
return S

all three sentences are used in the computation of the triplet loss. An important aspect of models based on any type of contrastive loss, including triplet loss, is the sampling methodology (Wu et al., 2017). For **BERT+PI** our triplet sampling algorithm involves randomly sampling examples based on label and the relationship between a hyperbole and its literal paraphrase, see Algorithm 1 and see Table 6 for examples.

The logic in our sampling algorithm is that if the anchor is a hyperbole, then we randomly sample another hyperbole as a positive (i.e., same class) sample for that triplet. We then set the negative sample to be the literal paraphrase of the positive sample (note: This sample is PI). This ensures that optimisation of the triplet loss forces a hyperbole to be closer to another hyperbole than its literal paraphrase in representation space.

If the anchor is *not* a hyperbole, we randomly sample a literal paraphrase as a positive sample for that triplet (note: This sample is PI). We then set the negative sample to be the hyperbole of the positive. The motivation here is that optimisation of the triplet loss will result in a non-hyperbolic sentence and a literal paraphrase being closer in representation space than a non-hyperbolic text and a hyperbole.

Formally, the class probability for an individual

Template	Example
{MASK}{MASK} is as {**JJ**} as {MASK}{MASK}	my heart is as **heavy** as the world
{MASK}{MASK} is {**JJR**} than {MASK}{MASK}	this version is **longer** than I expected

Table 5: **Quantitative Dimensions Test Examples**. ***Template*** shows templates as provided to CheckList, ***Example*** is an example sentence generated by CheckList.

Anchor	Positive	Negative
Inviting my mother-in-law to stay here is a **recipe for disaster**.	He eats a **mountain** of junk food.	He eats a *lot* of junk food.*
This supersonic airliner breaks the sound barrier.	Football is *important to him*.*	Football is **his oxygen**.

Table 6: **Semi-Random Triplet Sampling - Example Triplets**. ***Anchor*** indicates an anchor text. ***Positive*** indicates a positive text. ***Negative*** indicates negative text. Note: * indicates that the example is PI.

sentence is calculated by **BERT+PI** as follows:

$$\hat{y}_i = \sigma(e_i^a \mathbf{W} + b), \tag{1}$$

where e_i^a is the dense representation of anchor example i computed by BERT, $\mathbf{W}^Y$ and b^Y are learnable parameters and σ is a softmax function. The model is optimised via multi-task loss, see eq. 2.

$$\mathcal{L} = \mathcal{L}_c + \lambda \mathcal{L}_t \tag{2}$$

Where $\mathcal{L}_c$ is a binary cross entropy loss (eq 3), and $\mathcal{L}_t$ is a triplet loss (see eq. 4). λ is a parameter to weight the importance of the triplet loss and as a result the influence of the PI. In the cross-entropy loss, y_i is a binary indicator for class label, and $\hat{y}$ is the prediction output from eq. 1. In the triplet loss, D is the cosine distance, m is a hyperparamater indicating the margin, $e_i^a, e_{ij}^p, e_{ij}^n$ are the BERT representations for an anchor, positive and negative sample, and s is the sampling factor (i.e., how many positive and negative examples per anchor).

$$\mathcal{L}_c = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y_i}) + (1-y_i)\log(1-\hat{y_i})\right] \tag{3}$$

$$\mathcal{L}_t = \frac{1}{Ns}\sum_{i=1}^{N}\sum_{j=1}^{s}\left[\max(D(e_i^a, e_{ij}^p) - D(e_i^a, e_{ij}^n) + m, 0)\right] \tag{4}$$

5 Experiments

5.1 Baselines

We implement models presented in previous research on hyperbole as baseline methods for our experiments on hyperbole detection. Troiano et al. (2018) introduce an NLP pipeline style approach to detecting hyperbole in their foundational work on computational hyperbole detection. They introduce a number of hand-crafted features that are motivated by findings from cognitive linguistics on the mechanisms humans use for identifying and interpreting hyperbole. These features range from unexpectedness, imageability, polarity, subjectivity and intensity. These features are concatenated together and referred to as QQ (i.e., Qualitative and Quantitative) features by the authors, we adhere to that nomenclature and refer to our implementation of these features as QQ for the remainder of the paper. The authors experiment with several 'traditional' statistical learners for the classification layer of their pipeline. We use Logistic Regression and Naive Bayes, as those two methods were more accurate at the detection of hyperbole compared to the other methods in their experiments. We refer to these methods as **LR+QQ** and **NB+QQ** for the remainder of the paper.

Follow on from that work Kong et al. (2020) leverage the QQ features adjusting them slightly to compensate for differences in language and utilise pre-trained language models (i.e., BERT) for a hyperbole detection model. The authors combine the QQ features with the output from the BERT embeddings and pass the concatenated vector to a linear classification layer. We refer to this model as **BERT+QQ** in the remainder of the paper. We also include a simple vanilla BERT baseline that we refer to as **BERT** in the remainder of the paper.

5.2 Experiment Setup

We merge the Hyperbole Corpus and Minimal Units Corpus from **HYPO** and split into train-dev-test sets based on a 70:20:10 ratio. The Paraphrase Corpus is treated as a source of PI and thus only available at training time, also note that no sentences from **HyperProbe** were used for training,

Anchor	Positive	Negative
When the girl lost her puppy she cried an **ocean** of tears.	The little girl was **drowning in her** tears.	The little girl was crying a lot.*
I was crying for leaving my home.	My dad'll be very angry when he finds out that I wrecked his car.*	My dad'll **hit the roof** when he finds out that I wrecked his car.

Table 7: **Triplet Samples**. Examples of anchor, positive and negative samples generated by triplet sampler. Note: * indicates PI.

Hyperparameter	Values
Dropout	0.1, 0.2, 0.3
Learning Rate	1e-04, 1e-05, 1e-06
λ	0.25, 0.5, 1
s (Sampling Factor)	1, 3, 5
Encoder	BERT, RoBERTa

Table 8: **Hyperparameter search**. ***Hyperparameter*** indicates the hyperparameter. ***Values*** indicates the values used in search. Note: Not all parameters are applicable for all models (i.e., λ, s only required for **BERT+PI**)

.

Model	F1	Precision	Recall
LR+QQ	0.710(-)	0.679(-)	0.745(-)
NB+QQ	0.693(-)	0.689(-)	0.696(-)
BERT	0.709(.064)	0.711(.077)	0.735(.177)
BERT+QQ	0.671(.086)	0.650(.147)	0.765(.246)
BERT+PI	**0.781(.012)**	**0.754(.053)**	**0.814(.039)**

Table 9: **HYPO Results**. We provide the mean F1, precision and recall score as well as standard deviation across three runs for all models.

only testing. Overall we are left with four test datasets, HYPO, Extreme Case Formulations, Qualitative Hyperbole and Quantitative Hyperbole. We perform grid-search to find optimal hyperparameters for **BERT**, **BERT+QQ**, **BERT+PI**, see Table 8.

6 Results

6.1 HYPO

Results of our experiments on **HYPO** show that models that incorporate PI outperform the baselines, with respect to $F1$ score, see Table 9. We see a .071 (10%) increase in F1 for **BERT+PI** over the best performing baseline (**LR+QQ**). We use LIME (Ribeiro et al., 2016) to provide explanations for model predictions, see Figure 2. From this Figure we see examples that suggest that the increase in both precision and recall for **BERT+PI** seen in Table 9 is a result of a better contextual understanding of hyperbole-prone ECF terms. The first two examples in particular highlight the understanding of the word '*brainless*' in both a hyperbolic and non-hyperbolic context that are correctly classified by **BERT+PI**.

Model	F1	Precision	Recall
LR+QQ	0.678(-)	0.747(-)	0.621(-)
NB+QQ	0.523(-)	0.690(-)	0.421(-)
BERT	0.490(.340)	0.751(.158)	0.516(.453)
BERT+QQ	0.540(.337)	0.721(.184)	0.632(.484)
BERT+PI	**0.701(.014)**	**0.756(.033)**	**0.656(.047)**

Table 10: **Hyperprobe Results. Extreme Case Formulations**

Model	F1	Precision	Recall
BERT	0.407(-)	0.333(-)	0.522(-)
BERT	0.336(-)	0.400(-)	0.290(-)
BERT	0.278(.275)	0.240(.209)	0.401(.497)
BERT+QQ	0.352(.307)	0.255(.227)	0.599(.529)
BERT+PI	**0.527(.030)**	.486(.054)	0.590(.089)

Table 11: **Hyperprobe Results. Qualitative Hyperbole**

6.2 Extreme Case Formulations

From Table 10 we see models that incorporate PI provide improvements in detecting ECF hyperbole, .023 increase in F1, compared to **LR+QQ**. This aligns with results observed in Section 6.1 regarding the better understanding of hyperbole-prone ECF words in hyperbolic and non-hyperbolic contexts by **BERT+PI** compared to the baselines. We provide LIME explanations, (see Figure 3), and again observe examples that indicate a better contextual understanding of hyperbole-prone ECF terms by **BERT+PI**.

6.2.1 Qualitative Hyperbole

From Table 11 we observe that all models struggle to detect qualitative hyperbolic expressions, **BERT+PI** achieves the highest $F1$ of only 0.527 with a sub-0.5 precision of 0.486. With respect to variance we see many models with wild variances in recall, ($.529$, $.497$), suggesting that some of these runs are degenerating to outputting all positive class or all negative class predictions. These results suggest that qualitative hyperbole is harder to detect than ECF hyperbole.

BERT		BERT+PI	
LIME Word Weightings	**P(*h*)**	**LIME Word Weightings**	**P(*h*)**
Search engines are brainless entities.	.66	Search engines are brainless entities.	.18
Me, the wife of that boorish, brainless man.	.78	Me, the wife of that boorish, brainless man.	.74
This policy will plunge the country into a chaos.	.20	This policy will plunge the country into a chaos.	.79
Every flavor is dynamite.	.35	Every flavor is dynamite.	.96

Figure 2: **Model Explanation Comparisons - HYPO.** ***LIME Word Weightings*** indicate the importance of a word for a particular class, **orange** highlights indicate hyperbolic words, **blue** highlights indicate non-hyperbolic words. ***P(h)*** is the prediction probability that a sentence was hyperbolic with **red** indicating an incorrect classification (assuming a .5 decision threshold)

LR+QQ		BERT+PI	
LIME Word Weightings	**P(*h*)**	**LIME Word Weightings**	**P(*h*)**
the absolute majority was significant	.69	the absolute majority was significant	.35
the exam result was absolute	.70	the exam result was absolute	.12
the dead will never return	.53	the dead will never return	.02
nobody in the group looked interested	.51	nobody in the group looked interested	.10

Figure 3: **Model Explanation Comparisons ECF Tests.**

Model	F1	Precision	Recall
LR+QQ	**0.615(-)**	**0.5(-)**	**0.8(-)**
NB+QQ	0.565(-)	0.5(-)	0.65(-)
BERT	0.576(.048)	0.463(.001)	0.775(.177)
BERT+QQ	0.552(.183)	0.470(.073)	0.733(.379)
BERT+PI	0.590(.088)	0.492(.048)	0.750(.200)

Table 12: **Hyperprobe Results. Quantitative Dimensions**

BERT+PI	
LIME Word Weightings	**P(*h*)**
her brain is as small as a quarter	.86
Her hair is as thin as silk	.84
my heart is as heavy as the world	.73
his mouth is as big as a barn	.87
His beard is as thick as his mustache	.87
that bag is as heavy as a suitcase	.72
Her sister is as tall as her mother	.86
their hair is as long as a finger	.74

Figure 4: **LIME Explanations - Quantitative Dimensions**

6.2.2 Quantitative Hyperbole

From Table 12 we see that all models struggle to detect quantitative hyperbole and display a similar pattern of high recall (0.633 to 0.800) and low precision (0.463 to 0.5).

From an analysis of LIME explanations we identified one particular decision pattern as the source of many false positives. For sentences generated using the comparative sentence template (i.e., *{MASK}{MASK} is as {JJ} as {MASK}{MASK}*), the model always predicts a hyperbole irrespective of the comparison being made (see Figure 4). We observe that the first word of the sentence and the words and phrases '*is*', '*as*', '*is as*' and '*as a*' are the most influential words that lead to the decision to classify the sentence as a hyperbole. Our hypothesis for this error is that the literal paraphrases of hyperbolic expressions that take this form remove many tokens from the original sentence (e.g., '*He's **as** mad **as a** hippo with a hernia*' → '*He's **very** mad*'). We suspect this contributes to particular words and phrases (e.g., 'is as' and 'as a') being incorrectly considered hyperbolic because

they were removed from the original sentence during the literal paraphrase. We also note, that this is a particularly common form of hyperbolic expression in the training data (e.g., '*There lived a man **as** big **as a** barge*' '*He has **as** many debts **as a** dog has fleas*', '*He's **as** mad **as a** hippo with a hernia*'. '*you look **as** white **as a** ghost*').

7 Related Work

Troiano et al. (2018) posed the hyperbole detection task as a binary sequence classification task and introduced a dataset of annotated hyperbole as a benchmark for this task. The existing methods for detecting hyperbole, albeit scant, share similarities to methodologies for solving the problem of detecting other figures of speech. Generally, features are hand-crafted based on linguistic insights of a particular phenomenon (e.g., hyperbole) then combined with general purpose representations of textual content (Barbieri and Saggion, 2014; Joshi et al., 2016; Troiano et al., 2018; Abulaish et al., 2020). We see this in sarcasm detection (Joshi et al., 2016), irony detection (Barbieri et al., 2014) and metaphor detection (Jang et al., 2015). With respect to hyperbole, we see this approach in the foundation work on hyperbole detection (Troiano et al., 2018). Approaches to figurative language detection based on deep learning models have been also developed, such as irony detection (Huang et al., 2017), sarcasm detection (Ghosh and Veale, 2016) and metaphor detection (Wu et al., 2018). With respect to hyperbole detection, research has shown that deep learning improves accuracy on the task of detection of hyperbole in Mandarin Chinese compared to the use of traditional statistical learners (Kong et al., 2020). We extend upon both of these works by introducing a new model for hyperbole detection and introducing new data to evaluate hyperbole detection models.

Recent research in NLP, and machine learning in general, has focused on the idea of explainability and interpretability. The problem of understanding the reasoning behind decisions made by increasingly complex models on increasingly complicated data is a core challenge and can be a roadblock to research progress (Ribeiro et al., 2016, 2020; Bhatt et al., 2020; Linardatos et al., 2021). We design a suite of synthetic test sentences to probe the capabilities of hyperbole detection models and utilise the LIME framework(Ribeiro et al., 2016) for local explainability to understand the reasoning behind the decisions made by hyperbole detection models. Our approaches to probing and explainability are based on existing efforts to uncover meaning in decisions made by NLP models (Ribeiro et al., 2016, 2020; Rogers et al., 2020; Liu et al., 2021).

8 Conclusion

In this paper we proposed a hyperbole detection model, **BERT+PI**, that incorporates PI via triplet loss with a pre-trained language model (BERT) into a multi-task text classification framework for hyperbole detection.

Experiment results showed improvements in detection using standard information retrieval metrics (i.e., F1, precision and recall), for models that incorporate PI on the **HYPO** test set. However, these results were not maintained across our synthetic test suite **HyperProbe**. In fact, only on the ECF test in **HyperProbe** did we observe similar results. On both the quantitative and qualitative hyperbole tests we observed poor performance.

Our hypothesis for this disparity is that the incorporation of PI into **BERT+PI** teaches the model to learn annotation artifacts introduced by the creation of literal paraphrases in the Paraphrase Corpus of **HYPO**. Specifically, ECF hyperbole can often be paraphrased quite simply by removing only a few tokens (e.g., *what an **absolute** idiot* → *what an idiot*). **BERT+PI** effectively incorporates this information well and as a result appears to be able to differentiate between hyperbolic and non-hyperbolic ECFs. However, for more complex hyperbole, unwanted annotation artifacts are introduced during the process of creating a literal paraphrase. For example, '*my heart is as heavy as the world*' could be paraphrased as '*i am sad*'. In this paraphrase, the contrast and the semantic scale of the hyperbole are lost in the paraphrase given the significant difference between the hyperbole and the paraphrase. In future work, exploring better annotation methods for complex hyperbole that encode the semantic scale and the source of excessive contrast will be an important focus to overcome the shortcomings caused by unwanted annotation artifacts.

References

Muhammad Abulaish, Ashraf Kamal, and Mohammed J Zaki. 2020. A survey of figurative language and its computational detection in online social networks. *ACM Transactions on the Web (TWEB)*, 14(1):1–52.

Francesco Barbieri and Horacio Saggion. 2014. Automatic detection of irony and humour in twitter. In *ICCC*, pages 155–162.

Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014. Modelling sarcasm in twitter, a novel approach. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 50–58.

Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José MF Moura, and Peter Eckersley. 2020. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657.

Christian Burgers, Britta C Brugman, Kiki Y Renardel de Lavalette, and Gerard J Steen. 2016. Hip: A method for linguistic hyperbole identification in discourse. *Metaphor and Symbol*, 31(3):163–178.

Robyn Carston and Catherine Wearing. 2015. Hyperbolic language and its relation to metaphor and irony. *Journal of Pragmatics*, 79:79–92.

Claudia Claridge. 2010. *Hyperbole in English: A corpus-based study of exaggeration*. Cambridge University Press.

Derek Edwards. 2000. Extreme case formulations: Softeners, investment, and doing nonliteral. *Research on language and social interaction*, 33(4):347–373.

Aniruddha Ghosh and Tony Veale. 2016. Fracking sarcasm using neural network. In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 161–169.

Yu-Hsiang Huang, Hen-Hsen Huang, and Hsin-Hsi Chen. 2017. Irony detection with attentive recurrent neural networks. In *European Conference on Information Retrieval*, pages 534–540. Springer.

Hyeju Jang, Seungwhan Moon, Yohan Jo, and Carolyn Rose. 2015. Metaphor detection in discourse. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 384–392.

Aditya Joshi, Vaibhav Tripathi, Pushpak Bhattacharyya, and Mark Carman. 2016. Harnessing sequence labeling for sarcasm detection in dialogue from tv series 'friends'. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 146–155.

Li Kong, Chuanyi Li, Jidong Ge, Bin Luo, and Vincent Ng. 2020. An empirical study of hyperbole. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7024–7034.

Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. 2021. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18.

Leo Z Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A Smith. 2021. Probing across time: What does roberta know and when? *arXiv preprint arXiv:2104.07885*.

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*.

Michael McCarthy and Ronald Carter. 2004. "there's millions of them": hyperbole in everyday conversation. *Journal of pragmatics*, 36(2):149–184.

Lei Meng, Long Chen, Xun Yang, Dacheng Tao, Hanwang Zhang, Chunyan Miao, and Tat-Seng Chua. 2019. Learning using privileged information for food recognition. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 557–565, New York, NY, USA. Association for Computing Machinery.

Laura Cano Mora. 2009. All or nothing: A semantic analysis of hyperbole. *Revista de Lingüística y lenguas Aplicadas*, 4(1):25–35.

Neal R Norrick. 2004. Hyperbole, extreme case formulation. *Journal of Pragmatics*, 36(9):1727–1739.

Dmitry Pechyony and Vladimir Vapnik. 2010. On the theory of learnining with privileged information. *Advances in neural information processing systems*, 23:1894–1902.

Anita Pomerantz. 1986. Extreme case formulations: A way of legitimizing claims. *Human studies*, 9(2-3):219–229.

Sunny Rai and Shampa Chakraverty. 2020. A survey on computational metaphor processing. *ACM Computing Surveys (CSUR)*, 53(2):1–37.

Antonio Reyes and Paolo Rosso. 2014. On the difficulty of automatically detecting irony: beyond a simple case of negation. *Knowledge and Information Systems*, 40(3):595–614.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Yangyang Shu, Qian Li, Shaowu Liu, and Guandong Xu. 2020. Learning with privileged information for photo aesthetic assessment. *Neurocomputing*, 404:304–316.

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.

Kevin A Whitehead. 2015. Extreme-case formulations. *The international encyclopedia of language and social interaction*, pages 1–5.

Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. 2017. Sampling matters in deep embedding learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2840–2848.

Chuhan Wu, Fangzhao Wu, Yubo Chen, Sixing Wu, Zhigang Yuan, and Yongfeng Huang. 2018. Neural metaphor detecting with cnn-lstm model. In *Proceedings of the Workshop on Figurative Language Processing*, pages 110–114.