Laughing Heads: Can Transformers Detect What Makes a Sentence Funny?

Maxime Peyrard, **Beatriz Borges**, **Kristina Gligorić** and **Robert West**
EPFL
{maxime.peyrard, beatriz.borges, kristina.gligoric, robert.west}@epfl.ch

Abstract

The automatic detection of humor poses a grand challenge for natural language processing. Transformer-based systems have recently achieved remarkable results on this task, but they usually (1) were evaluated in setups where serious vs. humorous texts came from entirely different sources, and (2) focused on benchmarking performance without providing insights into how the models work. We make progress in both respects by training and analyzing transformer-based humor recognition models on a recently introduced dataset consisting of minimal pairs of aligned sentences, one serious, the other humorous. We find that, although our aligned dataset is much harder than previous datasets, transformer-based models recognize the humorous sentence in an aligned pair with high accuracy (78%). In a careful error analysis, we characterize easy vs. hard instances. Finally, by analyzing attention weights, we obtain important insights into the mechanisms by which transformers recognize humor. Most remarkably, we find clear evidence that one single attention head learns to recognize the words that make a test sentence humorous, even without access to this information at training time.

1 Introduction

Humor is a unique feature of human cognition and communication. The computational aspects of humor form a promising research area to advance the semantic understanding, commonsense reasoning, and language abilities of artifical intelligence (AI) systems. In particular, computational tasks such as humor detection and generation offer a rich and challenging testing ground for modern language understanding systems, to the extent that the ability to understand humor is commonly believed to be “AI-complete” [Stock and Strapparava, 2003].

Humorous texts often display complex narrative structures, rely on shared common-sense knowledge, and exploit subtle lexico-grammatical features of language [Raskin, 2008]. Nonetheless, impressive progress in humor detection has recently been made thanks to the advent of transformer architectures [Vaswani *et al.*, 2017] and the pretraining–finetuning paradigm [Devlin *et al.*, 2019]. For instance, Weller and Seppi [2019] finetuned a transformer to classify short texts as funny or serious, obtaining high accuracy on a dataset of Reddit jokes. They, however, neither performed error analysis nor gave insights into what signals are discovered and exploited by their model. In order to guide future research, a better understanding of how current models behave in practice, as well as a sharper outline of their capabilities, appears necessary.

In this work, we propose to leverage aligned pairs of funny and serious sentences collected via Unfun.me, an online game where players make minimal edits to satirical news headlines with the goal of making other players believe the results are serious headlines [West and Horvitz, 2019a]. Previously, West and Horvitz [2019a] released and analyzed 2.8K pairs from Unfun.me in order to better understand how humans create humor. Here, we use an extended dataset with more than 20K new pairs to probe the capacities of modern transformers such as BERT [Devlin *et al.*, 2019]. The minimal modifications between funny and serious headlines allow us to zoom in precisely on where the humor happens and thus perform entirely new kinds of analysis.

West and Horvitz [2019a] found that the difference between funny and serious headlines generated by humans tend to be explained by an opposition along certain dimensions important to the human condition, e.g., *reasonable vs. absurd* or *non-obscene vs. obscene*. We use the annotated, aligned pairs and the identified dimensions of opposition in order to perform fine-grained error analysis of transformer models.

Contributions. We finetune and evaluate standard BERT variants for two different humor detection tasks: (i) the *single-sentence setup*, where models detect whether an input sentence is funny or not, and (ii) the *paired setup* (cf. Sec. 4), where models detect for an input pair of aligned funny and serious sentences which one is funny. Based on the natural pairing of the data, we obtain diverse insights about the data and models:

1. In a careful error analysis, we characterize easy vs. hard instances (Sec. 4.2 and 5.1).
2. Inspection of attention heads reveals that the critical computation takes place in the last transformer layers, confirming that the model picks up on semantic, rather than lexical or syntactic, features (Sec. 5.2).
3. A particularly striking result is the emergence, during finetuning, of a head (the “laughing head”) specialized in attending to the funny part of an input sentence. This

head alone detects the humorous part of a sentence five times more accurately than a random baseline (Sec. 5.3).

Code and data,[1] as well as an extended version of the paper (with the appendices referenced here),[2] are available online.

2 Related Work

We discuss two aspects of previous research efforts on computational humor relevant to our work: datasets and approaches.

Humor detection datasets. Dataset creation approaches usually extract texts considered humorous online, e.g., on Twitter [Potash *et al.*, 2017; Zhang and Liu, 2014] or Reddit [Weller and Seppi, 2019], and, in parallel, collect serious texts from other sources to induce balanced datasets. For instance, Mihalcea and Strapparava [2005] collected 16K one-liner punchlines and matched them with 16K news headlines. Similarly, Yang *et al.* [2015] matched 2.4K puns with 2.4K headlines. Finally, the "Stierlitz" dataset [Blinov *et al.*, 2019] consists of 60K Russian jokes paired with news headlines. Others have directly collected pairs of funny and serious sentences. West and Horvitz [2019a] collected pairs via Unfun.me, an online game where players propose small edits to turn satirical news headlines into serious-looking ones. The Humicroedit [Hossain *et al.*, 2019] and FunLines [Hossain *et al.*, 2020] datasets were obtained via the reverse approach, where humans edited serious headlines into funny ones. The researchers have used their datasets to study humor perception and creation. Here, we use an updated version of the data collected by West and Horvitz [2019a] to study automatic humor detection methods.

Humor detection approaches. Researchers have applied humor detection techniques to several kinds of humor, e.g., irony [Wallace *et al.*, 2015; Reyes *et al.*, 2013], sarcasm [González-Ibáñez *et al.*, 2011], or satire [Goldwasser and Zhang, 2016]. Various baselines have been proposed, such as statistical *n*-gram analysis [Taylor and Mazlack, 2004] or classifiers based on human-crafted features [Purandare and Litman, 2006; Kiddon and Brun, 2011]. Following the evolution of the NLP field, pretrained word vectors lead to further improvements [Yang *et al.*, 2015; Cattle and Ma, 2018]. Finally, deep learning approaches have become ubiquitous [Chen and Soo, 2018; Blinov *et al.*, 2019]. In particular, transformers [Vaswani *et al.*, 2017] such as BERT [Devlin *et al.*, 2019] are now used to recognize funny sentences [Weller and Seppi, 2019] and generate funny texts [Horvitz *et al.*, 2020]Many researchers have aimed to understand how humans generate and perceive humor [Raskin, 2008]. Some of the resulting theories have been empirically verified. For instance, an analysis of the Unfun.me dataset [West and Horvitz, 2019a] produced evidence in favor of the *General Theory of Verbal Humor* [Attardo and Raskin, 1991]. The original and modified ("unfunned") headlines are generally opposed to each other along certain dimensions important to the human condition (e.g., *reasonable vs. absurd*, *high vs. low stature*, or *non-obscene vs. obscene*). On the contrary, despite impressive recent progress on automatic humor detection, the trained models have not been analyzed in depth. For instance, Weller and Seppi [2019] and Blinov *et al.* [2019] focus on reporting the performance of their transformer models. Yet, to further advance the field of computational humor and NLP in general, it is important to understand the capabilities of these models.

3 Data

To investigate humor detection using transformer models, we employ data particularly well-suited for a fine-grained understanding in controlled settings: pairs of funny and serious sentences with a small lexical difference collected via the Unfun.me game and previously released by West and Horvitz [2019a]. Unfun.me is an online game that incentivizes players to make minimal edits to satirical news headlines with the goal of making other players believe the results are serious news headlines.

We use an extended dataset of 23,113 pairs [West and Horvitz, 2019b],which we randomly split into 18,832 training pairs, 2,414 validation pairs, and 1,867 testing pairs. Additionally, as part of the game, a subset of pairs has been annotated with quality ratings measuring how well the unfunning process worked, i.e., whether the unfunned sentence was perceived as serious by other humans. These annotations come from other players who evaluated the quality of the unfunned sentences (for details, see West and Horvitz [2019a]). From these annotations, we form a restrictive *high-quality test set* of instances that received the maximum score according to all annotators, consisting of 754 pairs. We later refer to this test set as "HQ".

Finally, a subset of the test set (254 pairs) comes with manual annotations from two trained annotators capturing the opposition that leads to humor in the pair (cf. Sec. 1). In this work, we use the following seven types of opposition (listed with examples; satirical versions in bold; multiple oppositions may apply to the same pair; "$\emptyset$" refers to empty string):

1. normal/**abnormal**: *Bush picks* {***laser***, *rural*} *background for presidential portrait*
2. possible/**impossible**: *City opens new art* {***jail***, *museum*}
3. non-violence/**violence**: *Russian officials promise low* {***death***, *highway*} *toll for Olympics*
4. good/**bad** intentions: *BP ready to resume oil* {***spilling***, *drilling*}
5. reasonable/**absurd** response: *general motors reports record sales of new* {***disposable***, $\emptyset$} *car*
6. high/**low** stature: *Hollywood mourns passing of* {***16th or 17th Lassie***, *Robin Williams*}
7. non-obscene/**obscene**: *Tiger Woods announces return to* {***sex***, *golf*}

The advantage of such data for our analysis are threefold: (1) a naturally paired setup (2) with minimal modifications between funny and serious sentences and (3) additional annotations allowing us to investigate the performance of machine learning systems in fine-grained ways.

4 Models

We train standard transformer models to detect humor in the two setups described below.

[1] https://github.com/epfl-dlab/laughing-head

[2] https://arxiv.org/abs/2105.09142

Single-sentence setup. We first ignore the pairing between funny and serious sentences. The sentences are only associated with a binary label indicating whether they are funny or not. An encoder E maps a sentence s_i into a feature space, and a classifier converts the sentence representation $E(s_i)$ into a binary prediction. We train the model with the binary cross-entropy loss by finetuning the full pretrained model with backpropagation.

Paired setup. Next, we exploit the natural pairing of our data. Here, each funny–serious pair is first randomly ordered and then associated with a binary label indicating whether the first sentence is the funny one. This setup is naturally modeled via Siamese networks. For a given encoder E and an input pair (s_i, s_j), a classifier is trained on top of the concatenation of the feature representation of the two sentences, $[E(s_i), E(s_j)]$. Siamese networks have been successfully used recently in tasks based on the comparison of two sentences [Reimers and Gurevych, 2019].

Encoders used. The main focus of our analysis is on the BERT model, thus we employ BERT-base, an encoder with 12 layers and 12 heads per layer. Additionally, we report the performances of two BERT variants: distilBERT, a simplified and smaller alternative to BERT, and RoBERTa, a variant of BERT pretrained with an improved training setup. Each of these can be either used in the single-sentence setup (denoted "1S") or in the paired setup (denoted "PS"). For each setup, the BERT variants can be either fully finetuned or kept with weights frozen and only the classifier layer being finetuned. For reference, we also report the performance of baseline encoders without pretraining: BOW represents sentences by the average of their fastText word vectors [Grave *et al.*, 2018]; LSTM is a vanilla LSTM architecture trained on top of pretrained fastText vectors; and TRANSFORMER is a vanilla transformer with the same architecture as BERT, but trained from scratch without pretraining. Language models such as GPT2 [Radford *et al.*, 2019] are also important baselines, as it is often believed that humorous sentences are more surprising (i.e., have a lower probability according to the language model). In the paired setup, a natural baseline thus predicts the sentence with the lower GPT2 likelihood to be funny. In the single-sentence setup, a simple approach consists of predicting funny whenever the sentence probability is below some threshold previously identified via search on the training set. Later, we also use sentence probability scores provided by GPT2 in the analysis (cf. Sec. 5).

Error bars and significance level. Throughout the rest of the paper, error bars in plots represent bootstrapped 99% confidence intervals, and when we say that a change or difference is "significant", we technically mean that $p < 0.01$ in a paired t-test for comparing means.

4.1 Accuracy on Unfun.me Dataset

In Table 1, we report the accuracy of each encoder described above in both the single-sentence and the paired setup for both the full test set (*Full*) and the subset of the test set whose pairs were annotated as high-quality (*HQ*).

	1S		PS	
	Full	HQ	Full	HQ
No pretraining				
BOW	.511	.509	.515	.513
LSTM	.512	.511	.606	.598
TRANSFORMER	**.522**	**.526**	**.611**	**.607**
Pretraining, no finetuning				
GPT2	.526	.522	**.704**	**.682**
BERT	.536	.531	.689	.675
distilBERT	.534	.529	.685	.669
RoBERTa	**.575**	**.568**	.684	.675
Pretraining and finetuning				
BERT	.645	.641	.766	.737
distilBERT	**.651**	**.647**	**.777**	**.758**
RoBERTa	.649	.640	.755	.751

Table 1: Accuracy of various standard models for both setups: single-sentence (1S) and paired (PS). Datasets are balanced, so random baselines have accuracy 0.5. Best values per block in bold.

	1S		PS	
Type	GPT2	BERT	GPT2	BERT
normal/abnormal	.493	.630	.815	.849
possible/impossible	.518	.665	.790	.821
non-violence/violence	**.553**	.657	.842	.816
good/bad intentions	.532	.606	.702	.723
reasonable/absurd	.537	**.704**	.889	.907
high/low stature	.510	.647	.872	.892
non-obscene/obscene	.516	.582	**.918**	**.989**

Table 2: Accuracy per humor type as annotated in the Unfun.me dataset, for both the single-sentence (1S) and paired (PS) setup, and for finetuned BERT encoders and GPT2-based baselines.

The results indicate that humor detection is a challenging task, especially in the single-sentence setup: simple baselines such as BOW, LSTM, and TRANSFORMER barely improve upon random prediction. Only finetuned BERT architectures are significantly better than random, with the exception of RoBERTa, which achieves above-chance accuracy (57.5%) even without finetuning. In comparison, systems perform much better in the paired setup, where the classifier can take its decision based on the information from both sentences. The best model, distilBERT with finetuning, achieves an accuracy of 77.7%. Both BERT and RoBERTa achieve similar accuracy, with no significant difference. Overall, performance on the full test set and the HQ subset is similar, with a tendency towards slightly lower performance on the high-quality instances. However, the differences between the high-quality set and the full set are not significant, indicating that the full test set performance is a good indicator of performance on high-quality instances. Given the popularity of BERT and the similar performance of BERT, distilBERT, and RoBERTa, we focus on BERT encoders in the rest of the paper: BERT with no finetuning (simply called BERT), BERT finetuned on the single-sentence setup (BERT-1S), and BERT finetuned on the paired setup (BERT-PS).