Generating jokes with deep learning

Pietari Kaskela
604095
Aalto University
pietari.kaskela@aalto.fi

Abstract

In this paper we'll be looking at if neural networks can understand humour and generate jokes. Code available at https://github.com/eXeP/deepjokes.

1 Introduction

Natural language processing has been taking great strides recently with new architectures and ideas shaking the field. The field has recently started to shift away from recurrent neural networks towards transformer architectures, with almost every published state-of-the-art result now relying at least partly on these architectures.

Another trend is also moving towards more robust NLP-systems with larger models and more general datasets such as in GPT-2[8], ELMo [7], BERT [2] and ULMFiT [3]. These can be thought of as the "ImageNet" [5] of NLP, where tasks are not only trained on the task-specific dataset, but finetuned from a general model.

Despite these advances many more advanced and nuanced uses of language are still not solvable using NLP. One of these problems is the use and understanding of humour, which even for many people is completely unattainable. In this paper we'll be looking at how last-season models (RNNs) compare to a recent model (GPT-2) in generating jokes, and if either is capable of producing something resembling humour.

2 Dataset & Problem

For the data we'll be using Kaggle's Short Jokes-dataset [6] from 2017. The dataset contains approximately 232000 jokes scraped from various websites, but mainly from Reddit's /r/jokes-subreddit. The jokes are in English and of variable length between 10 and 200 characters. The dataset also contains 99 distinct characters and approximately 220000 distinct words.

So our problem now is how can we extract the essence of humour from this dataset and generate new never-before-seen jokes from it.

Table 1: Samples from the dataset.

September is Alzheimer's Awareness month... remind me tomorrow.
Well it's like my dad always told me ""When life gives ya lemons"" Chances are you're in the fruit aisle.
What English King invented the fireplace ? Alfred the grate !
I just saw the Assassins Creed Movie Trailer... I did not expect The Spanish Inquisition.
Why do Canadians do it doggy style? So they can both watch the hockey game.
Why does little sally have a limp? SHE WENT TO JARED!
""Oh you just put lotion on? You're not going anywhere."" - Doorknob
Why don't you want your nose to be 12 inches long? because then it would be a foot!

3 Methods & Implementation

We formulate our problem as maximizing the likelihood of a piece of text through the use of chain rule of probability:

$$P(\mathbf{x}) = \prod_{i=1} P(x_i|x_0, x_1...x_{i-1}) \tag{1}$$

where $\mathbf{x}$ is a sequence of words or tokens from a dictionary of size n. We then implement a neural network to approximate the function $P(x_i|x_0, x_1...x_{i-1})$ and train it with maximum likelihood (cross entropy loss).

So in order to generate new jokes from the trained neural network we feed it with some input (or a Start-of-Sentence token) and obtain some likelihood for each of the possible next tokens. We then must choose one of these tokens to be the next word in the sentence and repeat the process. Choosing a token is called sampling and different strategies for this are discussed in 3.5.

3.1 Recurrent neural networks

The first two of the models evaluated in this paper are recurrent neural networks (RNNs), which are class of neural networks suitable for sequences and time series. RNNs employ the use of a hidden state h_i to keep track of what has happened previously in the sequence. The hidden state is usually initialized to a vector zeroes and each iteration of the RNN takes as input one element of the sequence x_i and the previous hidden state, and produces as a result the output y_i and a new hidden state x_{i+1}.

3.2 Transformers

A more recent architecture proposed by [10] is the Transformer. It has recently reached state-of-the-art results on many tasks without relying on recurrence or convolutions. The transformer is of an encoder-decoder structure where the encoder maps an input sequence $(x_1, ..., x_n)$ to a sequence of continuous representations $\mathbf{z} = (z_1, ..., z_n)$ from which the decoder then generates an output $(y_1, ..., y_m)$.

The encoder consists of a stack of identical blocks, each with an attention head, residual connections and layer normalization and a standard feed-forward followed by a final residual connections and layer normalization. The decoder is almost the same with an additional attention and residual plus normalization block added at the start. More details on attention and implementation can be found in [10] and [8].

3.3 Embeddings

Instead of having the neural network input as ASCII-strings or one-hot-encoding we used learnable embeddings. The embeddings are a translation from the index of a token to a more suitable representation which carries information on the tokens relation to each other. The embeddings are implemented in Pytorch as n-dimensional real-valued vectors, where the similiarity of two vectors is measured by the cosine of the angle between the vectors (or dot-product). The embeddings can be learned during training using standard backpropagation.

3.4 Training

We split the dataset to training pairs $(\mathbf{x}, y)$, where x is a string of token indexes in the vocabulary $(x_0, x_1, ..., x_{i-1}$ and y is the index of the token x_i. We then used several different optimization algorithms such as vanilla SGD, ADAM, Adagrad and RMSprop to optimize the weights. We observed no differences between the optimizers.

3.5 Sampling

The output of the neural network (after softmax) can be seen as a multinomial distribution, where each token x_i has a probability p_i of being chosen as the next token. Contrary to what one might expect, it is not always beneficial to choose the token with the highest probability p_i, as this leads to repeating text. One other way to sample is to draw a sample from the multinomial distribution, but this also doesn't guarantee best results, as non-sensical tokens might be chosen. Following are descriptions of two sensible sampling methods.

3.5.1 Top-k sampling

In top-k sampling we construct the multinomial from only the top-k most probably tokens. This eliminates the possibility of non-sensical tokens being drawn and greatly enhances the quality of the text.

3.5.2 Top-p (Nucleus) sampling

Top-p sampling is a rather new sampling method from [1]. In top-p sampling instead of choosing a fixed number of k most probable tokens, we instead fix some probability p and choose the minimum number of tokens whose cumulative sum of probabilities exceeds p:

$$p' = \sum P(x|x_{1:i-1}) \geq p \tag{2}$$

In practice this means we select highest probability tokens whose sum of probabilities exceeds p. A new multinomial distribution is then created by scaling each of the chosen tokens probability by: $p_i = p_i/p'$ and drawing the final sample from that.

4 Results

4.1 Word-level RNN

We first trained a word-level RNN on the training data for approximately 8 hours on T4 GPU. The network operates on a many-to-one basis mapping several previous tokens to the next token. The architecture consists of mapping the word index to an embedding followed by a single LSTM-layer with dropout and concludes in a linear layer. We based the implementation on pytorch examples [**?**]. The dataset is quite challenging as without any preprocessing we end up with over 200000 tokens (different words).

Table 2: Samples from the word-level RNN, without top-k sampling.

Let's get something drunk phrase lamps. By first, but they're screaming, british asshole.
You want to borrow the cat's main product in your but never lost your pants.. it drives her clothes off it.
What did the women!!! say when he mixed up his ""OK Patty stool
There's no eye eye into a poll of the following great! Although not it a Without to Murray.
Japan and you never turned into a doesn't you wanna hear a joke about it I hear was the worst punchline
Have you ever heard the one about the New Year engine? No? Really? neither gets it?
What do you call some man that drives carrying cancer into his half Day? Wonderwall mini-me
man who doesn't wish he had an emergency relationship with reddit Ulysses have no buccaneer.

Table 3: Samples from the word-level RNN, with top-k sampling (40).

What happened to the guy with the wheel over the moon? A: He was a little boy.
You might have to tell me that it's an idiot only one to make my dog feel too bad you have a lot of more like.
How do black people get a divorce? Because they're always being able to finish a new race.
What do they call a Mexican you racist can only go at the bottom of a pool? Chicken
Why did the cow cross the road? Because I was afraid to get laid of the paper at the side.
What do you call a gay magician? A small wheel and a dirty bus and the owner for the job.
What kind of car do you have no legs call an angry cow? Beef jerky

From the tables 2 and 3 we can clearly see the difference the top-k sampling makes, as without top-k there are multiple errors in sentence structures such as missing uppercasing or weird quotation marks. The "jokes" generated however are not of particularly great quality and even the authors were unable to find an actually good joke.

4.2 Character-level RNN

We trained a character-level RNN on the training data for approximately 8 hours on T4 GPU. The network is similiar to 4.1. The character-level RNN has a much smaller dimensionality in the tokens as there are only 99, but manages significantly worse performance. This is likely due to bad hyperparameters or some bug in implementation as others have managed a much better performance on similiar datasets [4].

Table 4: Samples from the character-level RNN, without top-k sampling.

How many photo who apologials? They get with you everyone.
Why was Pret, barker? They meet someone she, it is imagine.
Why kind of encected music genis? Wait old.
Why are a milk tackle and Home Jews are eat? It todg out 2 lifes call the fence-on the flowers you're Garderdid.
Why are the gets partinurs? Appla about 1 is killed that way to stide loss up on elam boy.

Table 5: Samples from the word-level RNN, with top-k sampling (40).

What do the time supposidou rounced? They're president I called you to find hobout.
Why was the flipler blondme light so I define sick that you brove get? Pillowed a facite a tervice.
How do you make at liesting just now the be pig of his bucks? Thened punched one of a honess
ow do you be a vowe trap Joke Breaking? God, and agrees ownes meo crust.
Where did a phancrab school eat into the Stat faveric decision cardly the wish? Please's necksause in the ropa
[What do you call a high turing Sares out of deadle Mexican? Eatu

As we can see from 4 and 5 the generated text is littered with spelling errors and non-existent words. The network has learned some structures related to the dataset (question-answer), but the only joke we can find here is the quality of the output.

4.3 GPT-2

GPT-2 has many significant advantages over the previous two, as it's much more recent, larger and has been pretrained on a very large corpus. We're using the 345M model finetuned using Neil Shepperd's code [9].

Table 6: Samples from the character-level RNN, top-k sampling (40).

Chili's and a pig? One is a little fatty, the other is a little ham.
How do you start a conversation about sex? Nicely enough you just start it, they'll either agree or disagree.
Why does Santa Claus have such a big sack? He only comes once a year.
what does the emo say fly when its flying ? gloom dog i'll see myself out
How do you call a female scientist a slut? In the box.
""This is the ride that killed Jimmy."" - the slaughterhouse worker if I had to kill myself
I'm throwing a party for people who can't ejaculate Come hungry.
What album do Scottish vegetarian Pixies follow? Meet Mylish
I'm stuck in a room with Sir Mix-A-Lot and Stevie Wonder. I don't know which one to join.
How do we know that cookies are cookies? Because they are in crumbs.
It was about time we stopped adhering to the ""one size does not fit!"" mentality. Apparently not.

Table 7: Samples from GPT-2 345M, with top-p (0,9).

How did the Cop spy on the suspect? They used a silencer!
What do you call a fish with no eyes? A fsh.
walks into grocery store *stocks shelves with inferior goods *retires to toilet
How to prevent pregnancy if you are using a tampon. Layering on her is SPACING.
What is the most irresponsible restaurant you can go to? Fifth.
What do you call a group of people who defend keeping their surnames? Coaltesernameism
Two gay guys walk into a bar... One of them says ""I can't believe I just blew 30 bucks in there""
Why was the peanut crying? He was in a jam
Cheesy joke Cheddar
Why was the tomato blushing? Because he saw the salad dressing
What's the difference between a transsexual and a vagina? A vagina is easier to eat.

The output (tables 6 and 7) of GPT-2 is much better than the previous models'. The grammar and sentence structures are very good and the model seems to have learned several different joke-structures. Many of the outputs can also be considered real jokes, however they are not original and repeat several times in the dataset. We were however able to find several jokes which were not in the fine-tuning set, but the originality of these must also be questioned as GPT-2 pre-training data very likely contains jokes too.

5 Conclusions & Future

In conclusion we can say that NLP has taken huge strides in just a couple of years. These advances have brought models that can keep the context for very long lengths of text, but full comprehension of more advanced uses such as humour is still a ways off. This does not however mean that neural networks are entirely incapable of humour as seen in 4.3.

References

[1] Maxwell Forbes Yejin Choi Ari Holtzman, Jan Buys. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *ACL*. Association for Computational Linguistics, 2018.

[4] Andrej Karpathy.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[6] Abhinav Moudgil. Short jokes, Feb 2017.

[7] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proc. of NAACL*, 2018.

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

[9] Neil Shepperd. nshepperd/gpt-2, May 2019.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.