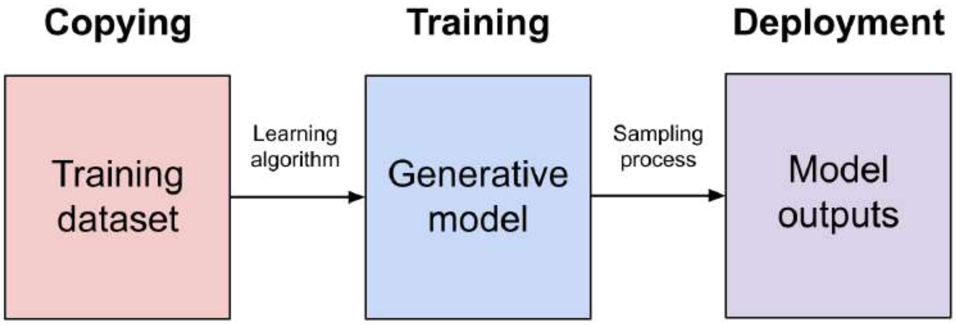
When does generative AI qualify for fair use?

Suchir Balaji
10/23/24



While generative models rarely produce outputs that are substantially similar to any of their training inputs, the process of training a generative model involves making copies of copyrighted data. If these copies are unauthorized, this could potentially be considered copyright infringement, depending on whether or not the specific use of the model qualifies as “fair use”. Because fair use is determined on a case-by-case basis, no broad statement can be made about when generative AI qualifies for fair use. Instead, I’ll provide a specific analysis for ChatGPT’s use of its training data, but the same basic template will also apply for many other generative AI products.

ChatGPT

Fair use is defined in Section 107 of the Copyright Act of 1976, which I’ll quote verbatim below:

Notwithstanding the provisions of sections 106 and 106A, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include—

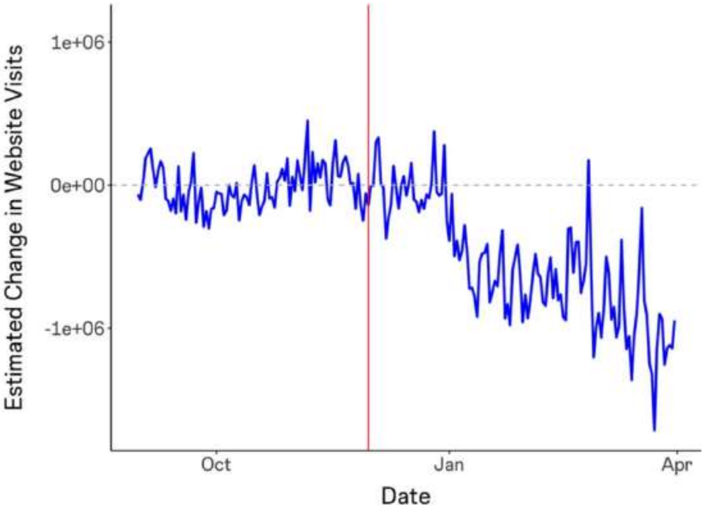
(1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
(2) the nature of the copyrighted work;
(3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
(4) the effect of the use upon the potential market for or value of the copyrighted work.

The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors.

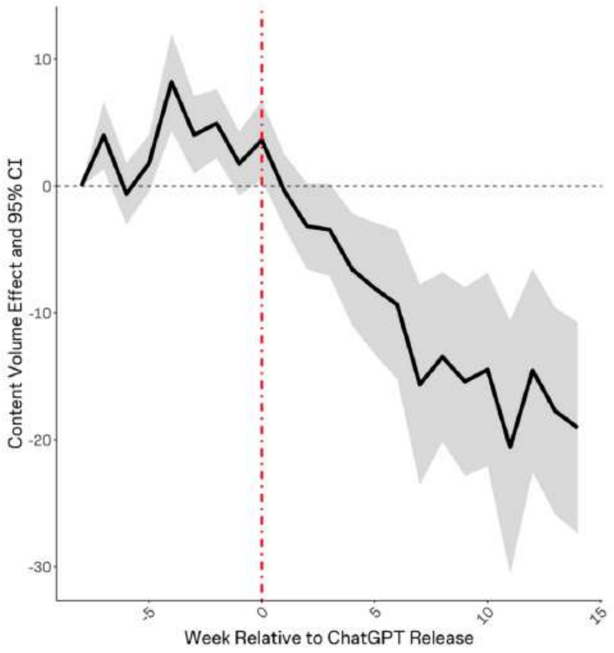
Fair use is a balancing test which requires weighing all four factors. In practice, factors (4) and (1) tend to be the most important, so I'll discuss those first. Factor (2) tends to be the least important, and I'll briefly discuss it afterwards. Factor (3) is somewhat technical to answer in full generality, so I'll discuss it last.

Factor (4): the effect of the use upon the potential market for or value of the copyrighted work.

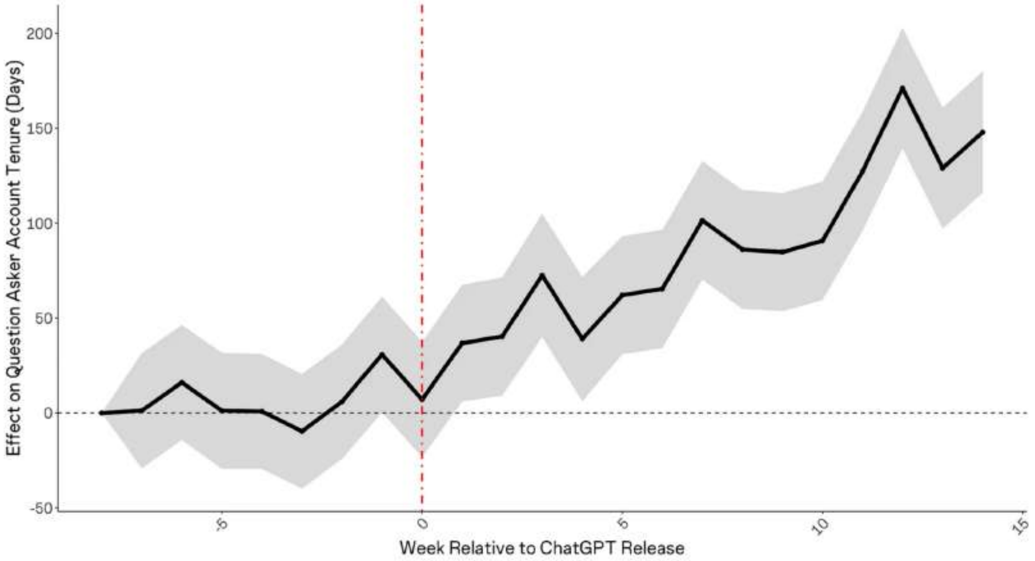
The effects on the market value for ChatGPT's training data are going to vary a lot source-by-source, and ChatGPT's training data is not publicly known, so we can't answer this question directly. However, a few studies have attempted to quantify what this could plausibly look like. For example, "The consequences of generative AI for online knowledge communities" found that traffic to Stack Overflow declined by about 12% after the release of ChatGPT:



They also find a decline in question posting volumes per-topic after the release of ChatGPT:



And lastly they find that the average account age of a question-asker trends up after the release of ChatGPT, suggesting that newer members are either not joining or are leaving the community:



These effects aren't going to be universal -- the same study didn't find similar declines in website activity on Reddit -- but it's unlikely that Stack Overflow is the only website affected by the release of ChatGPT. The homework-help website Chegg, for example, had its shares drop 40% after reporting that ChatGPT was hurting its growth. This isn't to say that ChatGPT was trained on Stack Overflow or Chegg, or even that the market effects on Stack Overflow and Chegg are a bad thing -- but there clearly can be market effects from ChatGPT on its training data.

Model developers like OpenAI and Google have also signed many data licensing agreements to train their models on copyrighted data: for example with Stack Overflow, Reddit, The Associated Press, News Corp, etc. It's unclear why these agreements would be signed if training on this data was "fair use", but that's besides the point. Given the existence of a data licensing market, training on copyrighted data without a similar licensing agreement is also a type of market harm, because it deprives the copyright holder of a source of revenue.

Factor (1): the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;

Taking from a copyrighted work and hurting its market value doesn't always disqualify from fair use. For example, a book critic could quote sections of a book in a critique, and while their critique might hurt the original book's market value, quoting it could still be considered fair use. This is because a critique has a different purpose than the underlying book, and so doesn't substitute for it or compete with it in the market.

This distinction -- between substituting and non-substituting uses -- is actually the origin of "fair use" from the 1841 case *Folsom v. Marsh*, in which the defendant copied parts of a biography of George Washington to make a version of their own. There it was ruled that:

> [A] reviewer may fairly cite largely from the original work, if his design be really and truly to use the passages for the purposes of fair and reasonable criticism. On the other hand, it is as clear, that if he thus cites the most important parts of the work, with a view, not to criticize, but to supersede the use of the original work, and substitute the review for it, such a use will be deemed in law a piracy.

In many recent cases, factor (1) has been considered in terms of "transformativeness" -- for example the Second Circuit's findings in *Authors Guild. v. Google* on Google Books that:

> Google's unauthorized digitizing of copyright-protected works, creation of a search functionality, and display of snippets from those works are non-infringing fair uses. The purpose of the copying is highly transformative, the public display of text is limited, and the revelations do not provide a significant market substitute for the protected aspects of the originals.

The Supreme Court has clarified the importance of "transformativeness" in the 2023 case *Andy Warhol Foundation for the Visual Arts v. Goldsmith*, noting that it should only be considered "to the extent necessary to determine whether the purpose of the use is distinct from the original", and that the first factor is "an objective inquiry into what use was made, i.e., what the user does with the original work". They note that "the first factor relates to the problem of substitution -- copyright's bête noire" and summarize it as follows:

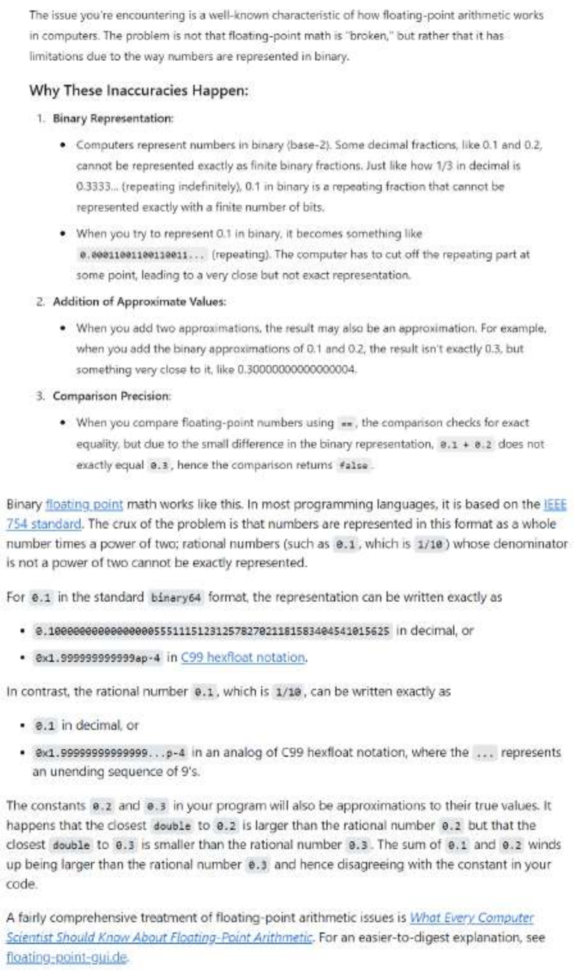
> In sum, the first fair use factor considers whether the use of a copyrighted work has a further purpose or different character, which is a matter of degree, and the degree of difference must be balanced against the commercial nature of the use. If an original work and a secondary use share the same or highly similar purposes, and the secondary use is of a commercial nature, the first factor is likely to weigh against fair use, absent some other justification for copying.

ChatGPT is a commercial product, so an initial question could be: does ChatGPT serve a similar purpose as its training data?

In practice, it's hard to reason about the "purpose" of a product as broad as ChatGPT, or the "purpose" of the entire internet. A better framing is: do the market harms from ChatGPT come from it producing substitutes that compete with the originals? Or is it an indirect effect, like a book critic could have on a book?

I think it's pretty obvious that the market harms from ChatGPT mostly come from it producing substitutes. For example, if we had the programming question "Why does

0.1 + 0.2 = 0.30000000000000004 in floating point arithmetic?”, we could ask ChatGPT and receive the response on the left, instead of searching Stack Overflow for the answer on the right:



The issue you're encountering is a well-known characteristic of how floating-point arithmetic works in computers. The problem is not that floating-point math is "broken," but rather that it has limitations due to the way numbers are represented in binary.

Why These Inaccuracies Happen:

1. **Binary Representation:**
    - Computers represent numbers in binary (base-2). Some decimal fractions, like 0.1 and 0.2, cannot be represented exactly as finite binary fractions. Just like how 1/3 in decimal is 0.3333... (repeating indefinitely), 0.1 in binary is a repeating fraction that cannot be represented exactly with a finite number of bits.
    - When you try to represent 0.1 in binary, it becomes something like `0.00011001100110011...` (repeating). The computer has to cut off the repeating part at some point, leading to a very close but not exact representation.
2. **Addition of Approximate Values:**
    - When you add two approximations, the result may also be an approximation. For example, when you add the binary approximations of 0.1 and 0.2, the result isn't exactly 0.3, but something very close to it, like 0.30000000000000004.
3. **Comparison Precision:**
    - When you compare floating-point numbers using `==`, the comparison checks for exact equality, but due to the small difference in the binary representation, `0.1 + 0.2` does not exactly equal `0.3`, hence the comparison returns `false`.

Binary floating point math works like this. In most programming languages, it is based on the IEEE 754 standard. The crux of the problem is that numbers are represented in this format as a whole number times a power of two; rational numbers (such as `0.1`, which is `1/10`) whose denominator is not a power of two cannot be exactly represented.

For `0.1` in the standard `binary64` format, the representation can be written exactly as

- `0.1000000000000000055511151231257827021181583404541015625` in decimal, or
- `0x1.999999999999ap-4` in C99 hexfloat notation.

In contrast, the rational number `0.1`, which is `1/10`, can be written exactly as

- `0.1` in decimal, or
- `0x1.99999999999999...p-4` in an analog of C99 hexfloat notation, where the `...` represents an unending sequence of 9's.

The constants `0.2` and `0.3` in your program will also be approximations to their true values. It happens that the closest `double` to `0.2` is larger than the rational number `0.2` but that the closest `double` to `0.3` is smaller than the rational number `0.3`. The sum of `0.1` and `0.2` winds up being larger than the rational number `0.3` and hence disagreeing with the constant in your code.

A fairly comprehensive treatment of floating-point arithmetic issues is *What Every Computer Scientist Should Know About Floating-Point Arithmetic*. For an easier-to-digest explanation, see floating-point-gui.de.

These answers aren’t substantially similar, but they serve the same basic purpose. The market harms from this type of use can be measured in decreased website traffic to Stack Overflow.

This is an example of an exact substitute, but in reality substitution is a matter of degree. For example, existing answers to all of the following questions would also answer our original question, depending on how much independent thought we’re willing to put in:

(1) “Why does 0.2 + 0.4 = 0.6000000000000008 in floating point arithmetic?”
(2) “How are decimals represented in floating point?”
(3) “How do floating point numbers work?”

Factor (2): the nature of the copyrighted work;

The nature of copyrighted work -- whether it’s a creative work that’s highly protected by copyright, or a factual work that’s mildly protected by copyright -- will vary a lot on the internet. But most data on the internet is protected by copyright to some degree, so it’s unlikely that factor (2) will strongly support “fair use”. In practice, this factor tends to be the least important anyway.

Factor (3): the amount and substantiality of the portion used in relation to the copyrighted work as a whole

There’s two interpretations of factor (3):

(1) The inputs of the model are full copies of copyrighted data, so the “amount used” is the entirety of the copyrighted work. This would weigh against “fair use”.
(2) The outputs of the model are almost never copies of copyrighted data, so the “amount used” is almost zero. This could potentially weigh in favor of “fair use”.

But interpretation (2) isn’t quite correct, because the purpose of copyright isn’t to protect the exact works produced by an author (otherwise, it’d be trivial to bypass by making small tweaks to a copyrighted work). What copyright really protects are the creative choices made by an author. Collage art is a simple example of this distinction: a collage artist won’t gain copyright protection for the underlying works they use, but they will gain copyright protection for the creative choices they made in the arrangement of those works.

Similarly, although the typical novel author doesn’t invent new words, they do still gain copyright protection for the choices they made in arranging existing words together. Every word in a novel is the result of a choice -- that is, a selection of one outcome from a range of possible outcomes -- and it’s the sum of all these choices that is protected by copyright. We can study these choices quantitatively using information theory.

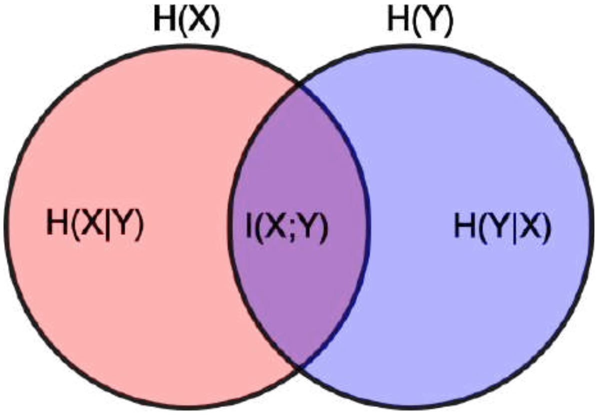
The unit of measurement for a single choice is the bit of information, representing one binary choice. The average amount of information in a distribution is the entropy of that distribution, measured in bits (Shannon first estimated the entropy of typical English text as being roughly between 0.6 and 1.3 bits per character). The amount of information shared between two distributions is their mutual information (MI), which can be expressed as:

$$I(X;Y) = H(Y) - H(Y|X)$$

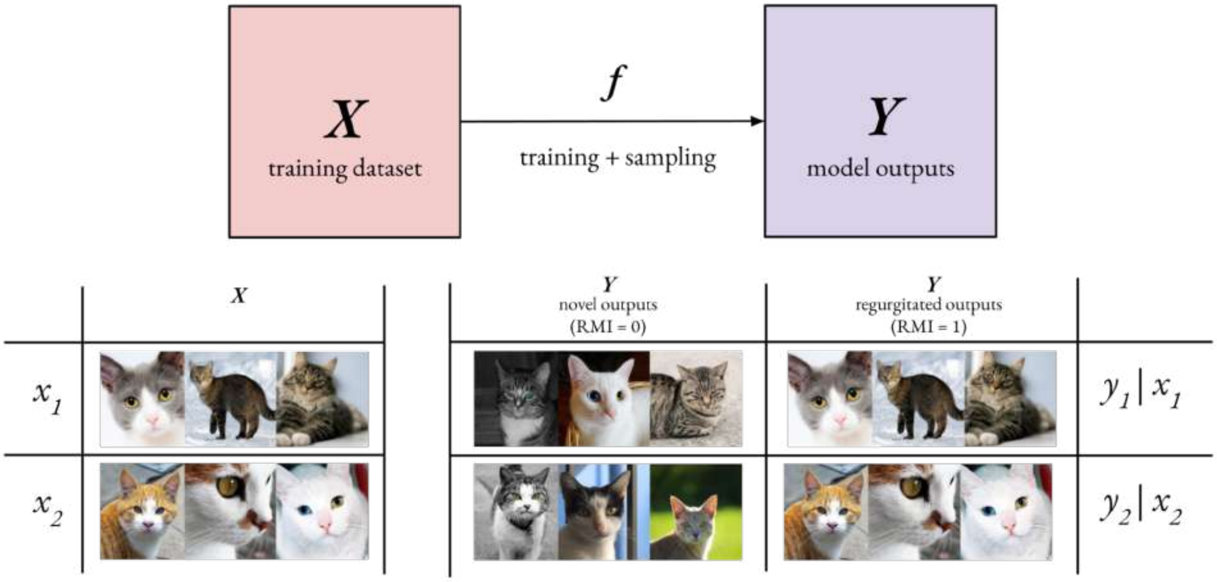
where X and Y are random variables, $H(Y)$ is the marginal entropy of Y, and $H(Y|X)$ is the conditional entropy of Y given X. If X is an original work, and Y is a transformation of it, then the mutual information $I(X;Y)$ indicates how much information from X was taken in creating Y. For factor (3) specifically, we care about the mutual information relative to the amount of information in the original work. We can call this the relative mutual information (RMI), and define it as:

$$I_R(X;Y) = I(X;Y)/H(X) = (H(Y) - H(Y|X))/H(X)$$

Visually, if the red circle below represents the information present in the original work, and the blue circle represents the information present in the new work, then the relative mutual information will be the area of the intersection relative to the area of the red circle:



In the context of generative AI, we are interested in the RMI where X represents a possible training dataset and $Y = f(X)$ represents a collection of model outputs, with f representing the process of training and sampling from a generative model:



In practice, it's usually easy to estimate $H(Y|X)$ -- the entropy of the outputs of a trained generative model. However, estimating $H(Y)$ -- the marginal entropy of model outputs aggregated over all possible training datasets -- will be intractable. Estimating $H(X)$ -- the true entropy of the training distribution -- is hard, but possible.

One assumption we could make is that $H(Y) \geq H(X)$. This could be reasonable to assume, because generative models that perfectly fit their training distribution will have $H(Y) = H(X)$, as will generative models that overfit and memorize their datapoints. Generative models that underfit may introduce additional noise, which could make $H(Y) > H(X)$. When $H(Y) \geq H(X)$, we can bound the RMI from below as:

$$I_R(X;Y) = (H(Y) - H(Y|X))/H(X) \ \geq 1 - H(Y|X)/H(X)$$

The basic intuition behind this bound is that low-entropy model outputs are more likely to be including information from the model's training data. In the extreme case, this is the problem of regurgitation, where a model deterministically outputs parts of its training data. But even nondeterministic samples can still use information from the training data to some degree -- the information may just be mixed in throughout the sample instead of directly copied.

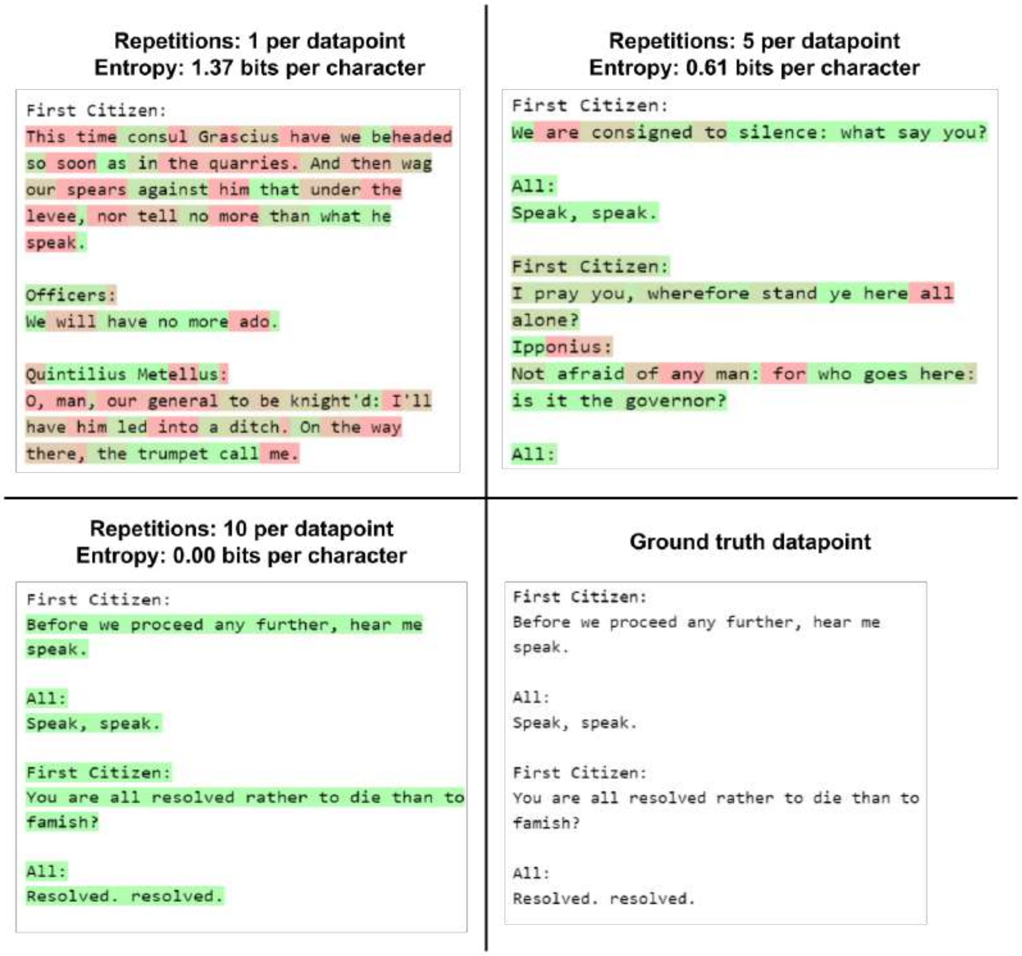
Note that there's no fundamental reason why the entropy of a model's outputs needs to be lower than the true entropy, but in practice model developers tend to choose training and deployment procedures that favor low-entropy outputs. The basic reason for this is that high-entropy outputs involve more randomness in their sampling process, which can cause them to be incoherent or contain hallucinations.

I'll list some entropy-reducing training and deployment procedures below, though this isn't an exhaustive list:

Data repetition during training

It's a common practice to show a model any particular datapoint multiple times during its training process. This isn't always problematic, but if done excessively, the model will eventually memorize the datapoint and regurgitate it at deployment time.

We can see a simple example of this from fine-tuning GPT-2 on a subset of the works of Shakespeare. The colors shown below indicate the per-token entropy $H(Y|X)$; red text is more random, and green text is more deterministic.



Repetitions: 1 per datapoint
Entropy: 1.37 bits per character

```
First Citizen:
This time consul Grascius have we beheaded
so soon as in the quarries. And then wag
our spears against him that under the
levee, nor tell no more than what he
speak.

Officers:
We will have no more ado.

Quintilius Metellus:
O, man, our general to be knight'd: I'll
have him led into a ditch. On the way
there, the trumpet call me.
```

Repetitions: 5 per datapoint
Entropy: 0.61 bits per character

```
First Citizen:
We are consigned to silence: what say you?

All:
Speak, speak.

First Citizen:
I pray you, wherefore stand ye here all
alone?
Ipponius:
Not afraid of any man: for who goes here:
is it the governor?

All:
```

Repetitions: 10 per datapoint
Entropy: 0.00 bits per character

```
First Citizen:
Before we proceed any further, hear me
speak.

All:
Speak, speak.

First Citizen:
You are all resolved rather to die than to
famish?

All:
Resolved. resolved.
```

Ground truth datapoint

```
First Citizen:
Before we proceed any further, hear me
speak.

All:
Speak, speak.

First Citizen:
You are all resolved rather to die than to
famish?

All:
Resolved. resolved.
```

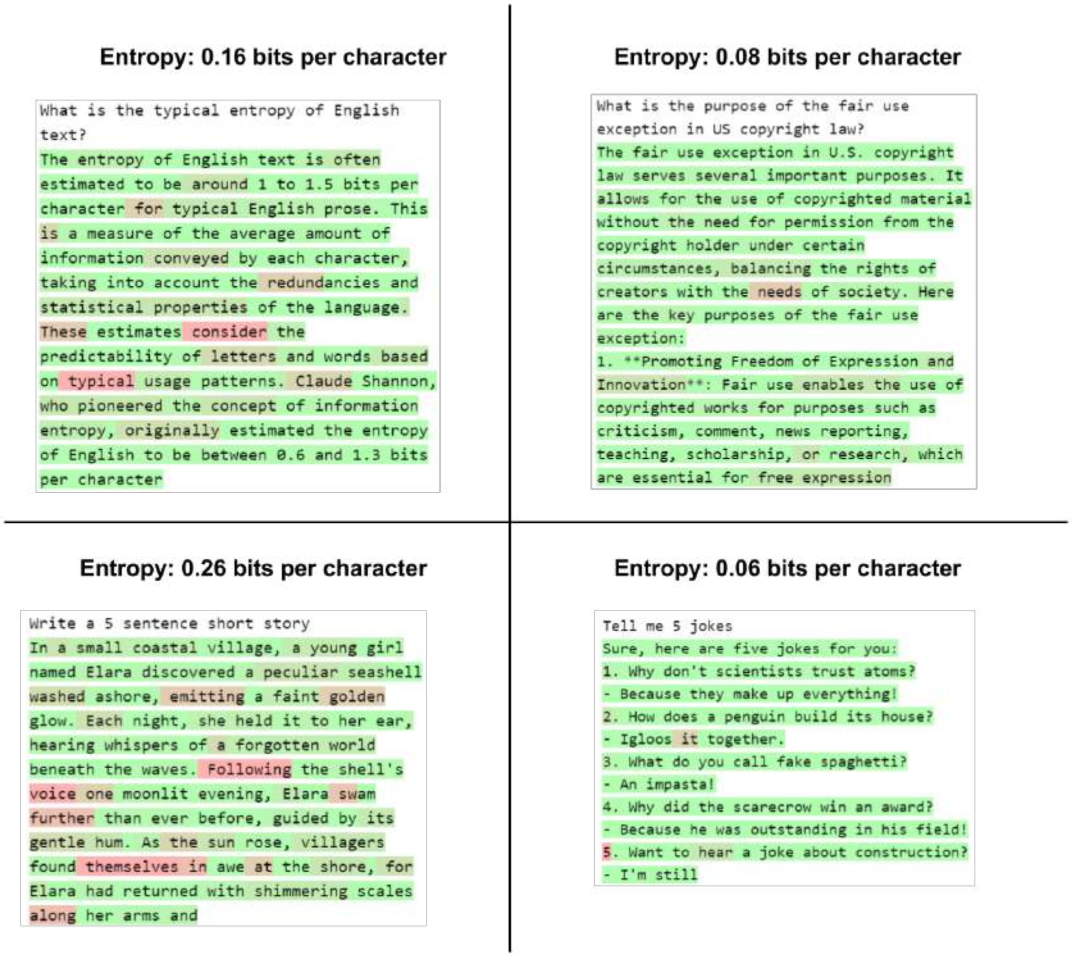
After the model trains on each datapoint once, its completions to the prompt "First Citizen:" are high entropy and novel, although incoherent. But after training on each datapoint ten times, it ends up memorizing the beginning of the play Coriolanus and regurgitating it when prompted.

At five repetitions, the model does something in between regurgitation and creative generation -- some parts of its output are novel, some are memorized, and the two are mixed together in its output. If the true entropy of English text was around 0.95 bits per character, we'd say that around $1 - 0.61/0.95 = 36\%$ of these outputs correspond to information in the training dataset.

Reinforcement learning

The main reason ChatGPT produces low-entropy outputs is because it is "post-trained" using reinforcement learning -- in particular reinforcement learning from human feedback (RLHF). RLHF tends to reduce model entropy because one of its main objectives is to reduce the rate of hallucinations, and hallucinations are often caused by randomness in the sampling process. A model with zero entropy could easily have a hallucination rate of zero, although it would basically be acting as a retrieval database over its training dataset instead of a generative model.

Below are a few example queries to ChatGPT, along with their per-token entropies:



Entropy: 0.16 bits per character

```
What is the typical entropy of English
text?
The entropy of English text is often
estimated to be around 1 to 1.5 bits per
character for typical English prose. This
is a measure of the average amount of
information conveyed by each character,
taking into account the redundancies and
statistical properties of the language.
These estimates consider the
predictability of letters and words based
on typical usage patterns. Claude Shannon,
who pioneered the concept of information
entropy, originally estimated the entropy
of English to be between 0.6 and 1.3 bits
per character
```

Entropy: 0.08 bits per character

```
What is the purpose of the fair use
exception in US copyright law?
The fair use exception in U.S. copyright
law serves several important purposes. It
allows for the use of copyrighted material
without the need for permission from the
copyright holder under certain
circumstances, balancing the rights of
creators with the needs of society. Here
are the key purposes of the fair use
exception:
1. **Promoting Freedom of Expression and
Innovation**: Fair use enables the use of
copyrighted works for purposes such as
criticism, comment, news reporting,
teaching, scholarship, or research, which
are essential for free expression
```

Entropy: 0.26 bits per character

```
Write a 5 sentence short story
In a small coastal village, a young girl
named Elara discovered a peculiar seashell
washed ashore, emitting a faint golden
glow. Each night, she held it to her ear,
hearing whispers of a forgotten world
beneath the waves. Following the shell's
voice one moonlit evening, Elara swam
further than ever before, guided by its
gentle hum. As the sun rose, villagers
found themselves in awe at the shore, for
Elara had returned with shimmering scales
along her arms and
```

Entropy: 0.06 bits per character

```
Tell me 5 jokes
Sure, here are five jokes for you:
1. Why don't scientists trust atoms?
- Because they make up everything!
2. How does a penguin build its house?
- Igloos it together.
3. What do you call fake spaghetti?
- An impasta!
4. Why did the scarecrow win an award?
- Because he was outstanding in his field!
5. Want to hear a joke about construction?
- I'm still
```

If $H(Y) = H(X) = 0.95\, bits/character$, we'd estimate between 73% to 94% of these outputs correspond to information in the training dataset. This could be an overestimate if RLHF makes $H(Y) < H(X)$, but there's still a clear empirical correlation between entropy and the amount of information used from the training data. For example, it's easy to see that the jokes produced by ChatGPT are all memorized even without knowing its training dataset, because they're all produced nearly deterministically.

This is a pretty rough analysis of how much copyrighted information from the training dataset makes its way to the outputs of a model, and an exact quantification of it is an open research question. But the higher order bit is it's non-trivial, so even the more generous interpretation of factor (3) would not clearly support fair use.

Fair use

None of the four factors seem to weigh in favor of ChatGPT being a fair use of its training data. That being said, none of the arguments here are fundamentally specific to ChatGPT either, and similar arguments could be made for many generative AI products in a wide variety of domains.