CS230

Final Report for CS230-Fall 2020
DeepWine

Andy Chow
andychow@stanford.edu

Jeff Sink
jeffsink@stanford.edu

Yu-Ann Wang
yuann@stanford.edu

Abstract

We used different sequence models, GPT2 and RNN/GRU, to generate wine descriptions/reviews. The finetuned GPT generates reviews that are considered valid reviews, but not necessarily good reviews for a particular wine. GRU yields less comprehensible results, as it does not come with the pre-packaged learning like the GPT.

1 Introduction

When we peruse wine selections online and in the store, we usually find them beautifully curated with in-depth descriptions from a reviewer (Wine Enthusiast, Wine Spectator, etc.) So in the case of an online wine retailer, which might have a catalogue of 10 million wines, it is very difficult to hire someone to generate description and content for every existing wine. The descriptions drive websearch traffic and thus sales.

This project works to generate descriptions of wines based on the characteristics of the wine. We do this by starting with a corpus of wine reviews, labelled by wine characteristics, to generate descriptions for wine that were not seen yet. We approach this through two very different models: RNN/GRU and GPT[2]. We have 3 goals 1) well-written text 2) a valid wine review 3) a review consistent with the wine at hand. The inputs (the characteristics) to our algorithm are [Price point, Rating, Grape Varietal, Geography].

2 Related work

Some work has been done to find the similarities between different wines. The early developmental work by Roald Schuring, put a lot of framework around how a wine can be described.[3] However, the majority of these projects take the description as an input and solve for a classification problem, the opposite of what we seek to do. In a previous CS230 project[5], the students used a bidirectional model to predict and generate reviews. The result looked to us to be obviously machine generated, and lacked common English sentence structure.

3 Dataset and Features

We leverage a ready to use data set from Kaggle[6]. The dataset contains 150K reviews of wines from all over the world, key columns include grape variety, points, region and the corresponding output, the descriptions. Formatting and processing was needed to fit the data into the GPT-2 model, including dropping 20,000 wines that did not have prices.

CS230: Deep Learning, Fall 2020, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

Field	Description	Summary	Example
Variety	The type of grapes used to make the wine	632 Varieties	Pinot Noir
Region_1	The wine growing area in a province or state	1237 Regions	Napa Valley
Price	Price of bottle in $	min $4 max $2300 med $24	$37
Points	Rating of Quality from WineEnthusiast	min 80 max 100 med 88	91
Description	A few sentences from a sommelier describing the wine's taste, smell, look, feel, etc.		This tremendous 100% varietal wine hails from Oakville and was aged over three years in oak

4 Methods

4.1 Simple RNN/GRU

For our initial experiment, we attempted to test our data with a simple RNN using Keras. Similar to the dinosaur language exercise in Coursera, we used our model to do character level prediction. Our initial dataset of 150K reviews was huge so we started with a batch size of 128. After indexing our characters, an embedding layer was applied, a GRU, and then a dense layer. For loss calculations we used a sparse categorical cross entropy function and Adam optimizer:

$$\sum_{c=1}^{M} y_{o,c} log(p_{o,c})$$

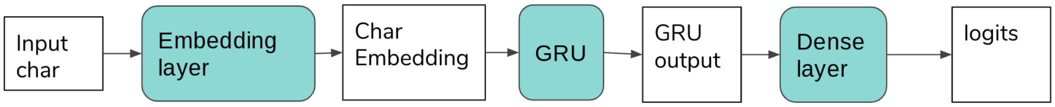

Figure 1: Initial architecture for the RNN/GRU

We attempted to train this over 30 epochs, but noticed (again probably due to the huge training data set) the loss started to increase around the sixth epoch. In our dataset, we had "seeded" each review with the country and varietal (see "country + varietal + description.txt"), and attempted to run the RNN with a "US Chardonnay" seed. Here are our initial results:

> us chardonnay here t's a bordeaux, with a deliciously repulated format, then shows nice acids and soft tannins with a port of old, the bouquet is tonic and compact on the palate and a crisp tang. peter of napa solid chardonnay (yes, this concentrated wine has lots of buoyant alcohol are in it, giving juicy acidity to flesh moderate impression.

It's far from comprehensible. In the following weeks, we continued to investigate different arrangements of layers (LSTM instead of GRU), using deeper networks, as well as a better technique of embedding wine concepts (`Doc2Vec` and GloVE) versus the character mapping to index we do as well as bi-directional RNN's to improve results.

In the meantime, we pivoted to GPT as a text generating model.

4.2 GPT as a generator

GPT-3 is a large text-in/text-out text transformer. It is famous for generating text that is hard to distinguish from human. But how to teach it about the structured data surrounding the wines, to allow it to make a good description of a specific wine? There are a few ideas in the literature. One usage of GPT extends it so it can detect the sentiment of a bit of text, we extend this idea here.[4]

GPT uses the cross-entropy loss function, but the GPT model generally hides that away from users. Instead users tune the model using parameters like `window` and `temperature`. They control the "randomness" of the the output text. The higher the temperature the more interesting the results, but with a tradeoff of perhaps higher probability of nonsensical results. Also users can `finetune` the model, which is basically a transfer learning application. It uses all the weights of the full GPT model, but then overlays it with weights to learn a specific task. The GPT is always an input-output model, so the `finetune` is also framed as input-output. We will describe our inputs and outputs below.

But we realize that the GPT-3 model is taking longer to train on our hardware. So we are using GPT-2, which will give use most of the benefits of GPT, but allows for faster iteration of ideas.

4.2.1 How to finetune GPT

We encode the structured data $(price, points, variety, region)$ as

- `price` -> `price_quintile`, because little difference between $56 and $55 wines.
- `points` -> `points_quintile`, because points (ratings) are subjective, and small differences are immaterial.
- price and points are each then replaced by strings `PRICE%s POINTS%s` where $s \in \{0, 1, 2, 3, 4\}$

We then make "prompts" (the inputs to the GPT model) that look like: (the model starts prompts at // and splits on the ||

- // `$points $price $variety $region` || `$description`
- // `POINT4 PRICE3 Chardonnay Napa Valley` || `This is a great little wine.`

A typical generated review:

> Very fresh and clean, with crisp, fresh green fruits, herbs, and a hint of tart green apples. It has all the right elements and gives a good overall score.

4.3 Measures of Success

As the project proceeded and our learnings became refined, we changed our measure of quality. What initially was thought to be challenging (producing content that looked human-generated) ended up being solved by GPT2, so we built a stairstep of more ambitious goals:

Goals	Description	Scorer	Metric
Human Written	Review is comprehensible, looks like it is written by a human	GLTR[1] (With GPT Option)	Review is "real" if the GLTR says that > 90% of the words are in the top-10 of the predicted values.
Looks like Wine Review	Text looks like a wine review (to a general human).	Human discriminator	Binary classification
Wine review appropriate for specific wine	Descriptors appropriately describe red vs. white wines	Subject matter experts/sommeliers in training aka our teammates	Binary classification

5 Experiments/Results/Discussion

5.1 RNN/GRU Experiments/Results/Discussion

To make our RNN model better, we tried several different tactics to improve the generation to look more human-written:

Table 1: Sample outputs for various RNN and params

Model/param	**Text generated**	**Loss**	**Human written?**	**Wine review?**	**Specific wine?**
GRU	"napa valley cabernet sauvignon has been so siz de m smells roasted oak, the finish backed by intense and delicate in its ruby color"	0.9129	N	N	N
LSTM	napa valley cabernet sauvignon, serothas ffe therd hed iricoat pla colaniso banthitspl	0.7586	N	N	N
Doc2Vec	california cabernet sauvignon from the producer's estate vineyards, this wine is about, and-like, with fruit flavors of and pomegranate, with a touch of white pepper and orange peel flavors .	N/A	Maybe	Maybe	N

5.2 GPT Experiments/Results/Discussion

The GPT model almost immediately gave nice looking results. But we still wanted to determine if the reviews were able to fool a detector like GLTR (with GPT option).

5.2.1 GPT Result: Good Wine Reviews are sometimes Bad English

In sending some of the wine reviews through the GLTR classifier, it tries to predict what the next word would be, given the previous words. If the next word is in fact in the top 10 (top-k) words, it is a good thing, and the text is likely to be valid.

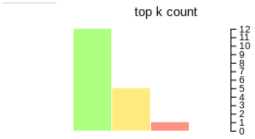

Figure 2: Green: words in top-10, Yellow: words outside top-10, Red: words outside top-100

But the wine reviews are not being classified as very good English because:

1. The vocabulary of the wine reviews involves a lot of rare words: oaky, tannic, cassis, etc.
2. The descriptions are often not full sentences, and read more like text messages on a phone

So we decided to NOT use the GLTR as our classifier. Instead we again used humans who would quickly see these were wine reviews.

The failure in GLTR is tragic, because it is the fault of the training data that the GPT faithfully learned. It learned the peculiar vocabulary of the wine reviews, and well as the abrupt style of online reviews, both of which are not well modelled in the basic GLTR (using un-finetuned GPT).

In the end a random sampling of 100 text generated reviews were all considered good reviews by an non-expert human.

5.2.2 GPT Error analysis: When good reviews are too good for learning

We then try to assess if the description is valid for the specific wine at hand (a wine with those specific characteristics).

Some of the reviews were general, and would apply to the wine:

The fruit is pretty, but the acidity overwhelms the wine's creamy character.

but others are clearly too specific (unless improbably also from "Superior Wines"):

> Not for aging, but with a solid core of cherries (this is an easy wine to drink now), this is not going anywhere. Drink now. Imported by Superior Wines.

After analyzing the ones that were clearly inapplicable to the wine we were generating for, we saw patterns. In the original reviews used for training, well-meaning reviewers put in details very specific to that wine. Such as its history, origin, or other details to distinguish it from other similar wines.

Ironically, it was actually that effort which is hurting us. It makes the knowledge learned from those reviews less applicable and portable to a similar wine with the same characteristics.

Frustrating that better wine reviews in training actually make for worse generated wine reviews.

5.2.3 GPT - Mitigations and other experiments

So the reviews are good, but not necessarily great for the wine it is meant to be generating for.

We then ran a few experiments to try mitigate the issues and arrive at better results. Ideas:

1. **Clean up the training data** Try to remove details that are very wine-specific. This was not feasible, and was not attempted.

2. **Split all the reviews up into sentences**: So all the reviews would be 1-2 sentences long. This was easy to do programmatically, and the idea was that if it generated shorter reviews, maybe the "very specific details" would be averaged away. The results were shorter reviews, very much like subsets of the other reviews. But that meant that some reviews were short and could be applicable. But others were short and obviously wrong: `"A blend of 10 Côtes du Rhône, with 5% Grenache..."`. On balance, we thought it better to have longer reviews so that each one looks richer, with a chance of being somewhat wrong, better than a 5% of being completely wrong.

3. **Try varying the "keys", the data we use to characterize the wines**: Switched from `$points $price $variety $region || $description` to `$points $price || $description` or `$variety $region || $description`, but they all suffered roughly the same problems, but in general the more data in the key the better.

4. **Use quintiles for input price and points**: The model was not learning prices or quality scores very well, each one was in integers (some very high!), so we switched to `price_quintile` and `points_quintile`, and this allowed the model to better port the knowledge about one family of wines to another.

5. **Lower the temperature of the generated output**: **THIS WORKED!** We ask it to generate output with lower `temperature`, it decides to be more conservative in its choices, and thus avoid the details that were specific to one wine in the training data. The results were somewhere repetitive, fixating on a few adjectives, but much better overall. **The reviews were less colorful, but less wrong.**. Example: "It will age well over the next 4–5 years. Drink this to drink up."

Table 2: Results, varying temperature

Summary	**Original**		**Temp=0.9**		**Temp=0.6**	
	Count	**%-age**	**Count**	**%-age**	**Count**	**%-age**
Total descriptions evaluated	66		59		60	
Total descriptions accurate for cab	34	51%	33	56%	57	95%
Descriptions mentioning other varietal	13	20%	12	20%	0	0%
Descriptions which are just wrong	19	29%	14	24%	3	5%

6 Conclusion/Future Work

We found that building on top of the GPT model gives us the best outcome. The reviews are human readable and actually mean something. After fine-tuning the model with the Kaggle Wine-review data[6], the description generation is passable for a wine review. With the temperature lower, we are able to get a good and relevant review of a particular wine, without it going too far astray.

With more time we would explore more options to fine-tune the model to make it more specific but stay accurate. Other directions would include looking at using other wine characteristics (like Country) to see if it helps generate better reviews.

Further work is needed in order to make DeepWine usable for wine drinkers in the world.

7 Contributions

Jeff: GPT-2 model and fine-tuning with integration
Yu-Ann: RNN/GRU model
Andy: Data preparation and GloVe exploration

References

[1] S. Gehrmann, H. Strobelt, and A. Rush. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy, July 2019. Association for Computational Linguistics.

[2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. 2019.

[3] R. Schuring. Wine embeddings and a wine recommender. `https://towardsdatascience.com/robosomm-chapter-3-wine-embeddings-and-a-wine-recommender-9fc678f1041e`, May 2019.

[4] spronkoid. Gpt2-sentiment-analysis. `https://github.com/spronkoid/GPT2-sentiment-analysis`, Dec 2019.

[5] C. M. Tim Aiken. Applying natural language processing to the world of wine. `http://cs230.stanford.edu/projects_spring_2018/reports/8290440.pdf`, June 2018.

[6] zackthoutt. Kaggle wine reviews. `https://www.kaggle.com/zynicide/wine-reviews`, June 2017.

A Appendix: Sample outputs from GPT, TEMPERATURE = 1.0

```
Generated for PRICE3, POINTS3, Napa Valley, Cabernet Sauvignon

 A dark, dark, ripe wine that shows some ripe fruit, herbs, prunes and
spices and a smooth, elegant texture.  Drink this as soon as you can.  ||

This is the second season of the bottling.  The nose is fresh, fresh, as
it shows grapefruit, citrus and an almond note.  The palate is clean and
crisp with bright orange flavors.  Drink immediately.

This young, fruity wine is ready to drink.  Lemon peel and red berry aromas
are joined by a light touch of mineral to provide an even playing surface.
The wine shows considerable finesse and finish.

This is an excellent, complex wine, very dry and very well balanced with
plenty of fresh acidity.  The palate shows intense citrus, pine syrup,
melon and pear flavor, wrapped into crisp, bright acids.  Drink from
2016.
```

B Appendix: Sample outputs from GPT, TEMPERATURE = 0.6

```
Generated for PRICE3, POINTS3, Napa Valley, Cabernet Sauvignon

 A bit too soft for balance, but still a good value.  It’s a rich,
full-bodied wine, with flavors of red cherries, blackberries, currants,
bacon, sweetened vanilla and meringue.  It’s a bit on the sweet side, but
it finishes clean and dry.

A little hard and heavy in the mouth for the price, but it’s a good value.
It’s a light, fresh wine, with flavors of red cherries, red currants, red
currants and spices.  It’s easy to like, and a little on the sweet side.

This is one of the best of the winery’s new Signature Cabernet line.  It’s
dry and tannic, with a bit of a caramelized, caramelized note to its cherry,
cherry, currant and spice flavors.  The wine is very tannic, but there’s a
good chance of it being better in a few years.
```