Classification and Creation From Modeling Song Lyrics

EECS 349: Machine Learning June 2018 Northwestern University

Sabreen Ali sabreenali2019@u.northwestern.edu
Michael Benimovich michaelbenimovich2019@u.northwestern.edu
Stacey Chao staceychao2019@u.northwestern.edu
Charlie Collar collar@u.northwestern.edu

Part I: Introduction and Task

It is a common judgement on genres of music that their lyrics are all the same. But is that really the case? Given just lyrics to a song, would it be possible to train a model to identify which genre it belongs to? This task can be important for any sort of large database of lyrics, whether on a lyrics website or an online music store, to quickly identify the genre of song based on lyrics. This task could also be used for analyzing how similar genres actually are, based on how difficult it might be to classify songs only on their lyrics. This task is explored in Part III.

A second task, perhaps more interesting, related to song genre and lyrics is to see the possibilities of generating lyrics based on a model trained on songs of the same genre. That is, can we synthesize lots of country song lyrics and create a model that gives us quintessential (or cliché) song lyrics? This task is explored in Part IV.

Part II: Data

Data was scraped from metrolyrics.com. A simple PHP scraper was created to first gather all the artists based on a list of artists metrolyrics offers. Then, from each link to each artist, the lyrics to the top 65 songs (or fewer) of that artist were added to the dataset. In total, we gathered over 400,000 songs. In addition to taking the song lyrics and genre, we also took features for artist popularity (determined by metrolyrics when scraped on May 9th, 2018,) song title, artist, and year released.

We then filtered our dataset. We first eliminated songs from artists who were not popular, artists whose popularity was under 10 out of 100. We also eliminated songs from the dataset that did not specify a genre or were labeled as "other," as this genre label seemed to be a catch-all for songs that were either of niche genres, and we believed this would add unnecessary noise to our dataset, or possibly cause models to label an unusually high percentage of songs as "other."

We still had some problems with the dataset. The first issue was attempting to remove songs that were not entirely in English. While we used the Python library "langdetect" made by Google to filter songs that were not in English, but this only filtered songs that had no English; many songs were half English and another language, and these were not filtered. We were also concerned with the lack of different genre labels. One of the reasons we chose metrolyrics over other popular lyrics websites was because of the diversity of labels, although it is not as diverse as we would have liked.

Part III: Classifying Genre from Lyrics

We were aware that there would be problems with overfitting the data due to the massive amount we were able to collect, so we started with small samples and grew them until we were plateauing. All classification models and filtering was done using Weka.

We preprocessed the data by first eliminating all attributes besides genre and lyrics. We decided our task should be focused on lyrics, and wanted to create a classifier that was forced to work with minimal amounts of data and work with instances where year, popularity, etc. were not provided. We then eliminating all punctuation, leaving just alphanumeric characters.

We then used Weka's StringToWordVector filter to create numeric attributes from our string data. We adjusted some settings from the default, utilizing an IDF-TF transformation to get a better sense of word importance. The attribute fields were limited to about 1000 attributes, and we used Weka's built in Rainbow stopword handler (from Rainbow).

We analyzed three different classifications, Naïve Bayes, Random Forest, and Support Vector Classifier (Weka's SMO). We also used ZeroR and 1 Nearest Neighbor as control. For each classifier, we trained on 70% of the data and tested on the other 30%.

Starting with just 100 instances, almost all the results were the same, and quite low. When the dataset was around 300, we started to see more accurate results, as the ZeroR classifier started to plateau, thus was finally acting more as a control. At this small dataset, Naïve Bayes was performing very well, with over 50% accuracy. The other two classifiers were also performing well, with Random Forest at exactly 50% and SMO just under 50%. 300 and 600 were the only datasets were all three classifiers were performing better than ZeroR.

Once the dataset grew to be over 1,000, Naïve Bayes started to overfit, and its accuracy was continuously lower than ZeroR for every test greater than 1,000. SMO also started to overfit, although not quite to the same degree as Naïve Bayes. Interestingly, SMO started to do better again later on, when the dataset sizes grew to over 10,000, but Weka unfortunately did not have enough resources to run SMO on 20,000.

Data Size	ZeroR	IBk	Naïve Bayes	Random Forest	SMO
100	23.3333	23.3333	23.3333	23.3333	26.6667
200	36.6667	11.6667	40.0000	38.3333	43.3333
300	45.5556	45.5556	51.1111	50.0000	48.8889
600	38.8889	34.4444	40.5556	45.0000	42.2222
1,000	44.3333	39.0000	37.3333	48.3333	41.3333
1,500	43.5556	27.6667	38.8889	48.2222	38.0000
2,000	42.3756	35.6667	37.2392	47.0305	38.6667
3,000	42.0000	32.3333	33.7778	49.2222	39.8889
5,000	41.2000	34.9333	33.6000	47.4667	36.1333
10,000	42.5000	28.0333	32.6333	51.5333	40.2333
20,000	42.5333	27.7000	32.3167	51.9667	--

Table 1: Performance of classifiers (accuracy percentage) on varying dataset sizes

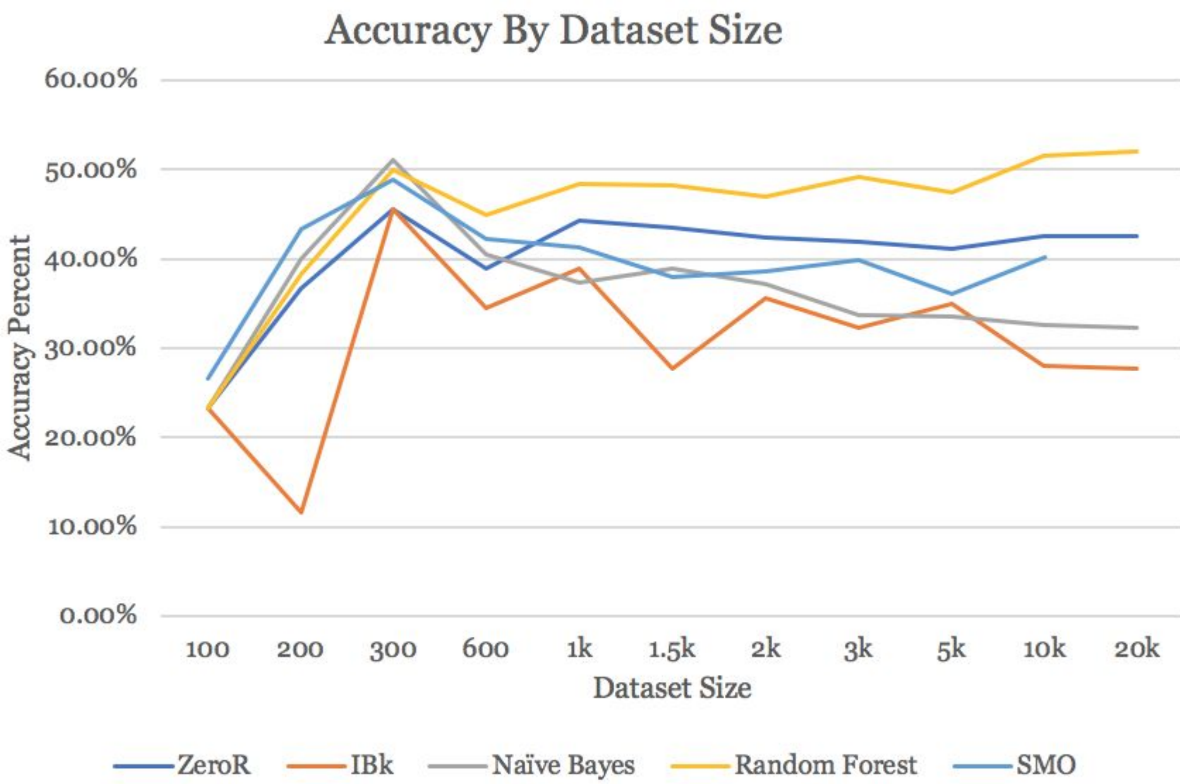


Figure 1: Performance of classifiers on varying dataset sizes

The only classifier that continued to do well was Random Forest, for a number of reasons. The first is its resistance to overfitting. Random Forests build an ensemble of smaller trees, then vote between the trees when classifying data. This is particularly good for text classification, as representing the strings as word vectors creates a massive amount of attributes which can be incredibly noisey.[1]

It was surprising that all our classifiers failed to obtain much more than 50%; this is below what we were expecting. This could be because the classification of genre by metrolyrics is not accurate, and we unfortunately have no way to ensure this accuracy. However, this could also be due to songs not having as cliché of lyrics as expected, and perhaps the genres are not as different as we thought. A better dataset is essential for future work, as it is uncertain if the poor results come from a bad dataset, or just the difficulty of the task.

Part IV: Generating Song Lyrics

For the second task of generating our own lyrics of a song given a genre, we used a TensorFlow Char-RNN, a character level language model using multilayer Recurrent Neural Network (source). Using the scraped lyrics data, we created one large text file (~600 kb) for each genre that contained a sample of the lyrics from that genre, about 300 songs from each genre. The five genres we chose to run for the project were: country, indie, hip hop, pop, and rock. We then trained a char-rnn on each of the those genres, and once trained, created a sample text of about 1000 words.

Each training parameter for all the genres were the same. We used a batch size of 100, 10 epochs to train, and didn't use a dropout.

After training, we then used the models we created to generate a sample set of lyrics for each of the genres. For generating each sample, we set the output length to be 1000 and tried various temperatures. The temperatures that were closer to 1 tended to be less coherent in both vocabulary and grammar, whereas the temperatures that were closer to 0 tended to be coherent but contain a lot of repetition. We found that the most accurate temperature overall, balancing coherency and repetition, was 0.5.

Each of the models we created generated the same general results; we had validation perplexities from around 4.5 - 6, and test perplexities which were slightly lower by a margin of 0.5. In general, this meant that our model had higher accuracy on the test data, which meant that our model in general did well on the test sets. The only exception to that was the model we generated for rock; for that model the best validation perplexity was 4.76, while the test perplexity wasa 5.35. This discrepancy could have likely occurred because the random sample of data we used for training did not contain as much variety as the other genres, and therefore the vocabulary distribution was lower. As a result, the accuracy on the training set was lower, leading to a higher perplexity, and leading to a lower quality sample that the other genres.

In the appendix, training results, the learning curve, and other model details can be seen for the models produced on Country, Hip-Hop, Indie, Pop, and Rock genres.

Part V: Conclusion

This task turned out to be harder than expected; as stated above, we thought it would be pretty easy to classify a song into a genre purely based on lyrics, but this result shows us that the music plays a much more substantive part (and perhaps, the only part) in our determining of genre. Furthermore, in the task of creating lyrics, temperature was difficult to determine; it was hard to find a balance between chaos and too much repetition, as both can be seen in the results.

Michael Benimovich created the website, and created and executed data preprocessing and filtering. Stacey Chao and Sabreen Ali created and executed all tests and experiments with preprocessing for the text generation. Charlie Collar created and executed data scrapers, and created and executed all experiments with text classification.

Appendix

Sample Country

I'm gone
Well my heart were was a blue
He wonder did you feel
I'm not in at finally come on home
But the road her round of lust
Shot about the world
Then the roor mama ten of life
How honey you when I'm gone
The land of your love
Strangers on the tears and take me up in the rolling called the blue
But you feel your love with a country song
I want to close
I've got to go
I want to me
The way your kinged to my soul
I want to be
The put your eyes
The stars and I'm gone
I've got a had a long that love
We was looking down the bord of last
I've got the seed
I don't really don't you frimedy
Well my brues about is a million
how I was a crowd
And he said
I could you rusty rolling down the world
You won't never going outta style
I want to tell you my family falling driving
I want to wake you had to tear you
I was all to so her
Baby, it's only you don't day
The world to stand you look
But I could you rock out
When I know you don't wan

Sample Hip-Hop

I'm so many get you
And I toush that we see that ould stay
I can tell by your hot that we say it's a vibe
It's this 1 thing low we get him
We can tell me where would I be
Oh baby, oh
Oh oh
Chorus
It's a vibe, yeah, yeah
Yeah, that's wanna sit it on my neck
And we had it
I am a big booty money
I can't get dangerous
I ain't got no bested one of she got me the sing
Fuck that bitch
I think it's the parking like Sprister
Watch out lil bitch
'Cause you can't go back
It's this 1 thing that my dog, that my dog, that my dog, that my dog, that my dog, that my dog
It's this 1 thing not don't love you
That my dog, that my dog, that my dog, that I feel good
I didn't have me that man
My dance with me
It's this 1 thing and I ain't got a problem
What can't was the pates
You did it feel so good
Let me do it like me, oh motherfuckin' me
So won't you dance with me
And I real chest and gave me
And I keep it me stay
We can't want a fuck and the style

Sample Indie

I'm the way
They want to find you
How would you called

The stars she was a stars
And I have alone in the world
I will save me and the best
I can't let that I was the s
I can tell in the wall
I
I want to be and the storms
I wish i have a child
If there another sun
What is though it
And so long time that you make me
It can you are gonna be the light
I into an every heart and the first time
I will never know
I have seen the light
I have always feel
So long in the street,
I can tell you
I'm waiting out of the sun
In the sea of love
I can't get out of the streets
I will gave a blink
And
Can I saw behind
That I go thinking that I can save the one with a bells
I was a song hard to say
As I want to be the people on the way
I am a part of my friends
When you're a vicion
I will gat something that you want
I'm gonna stay to the rain
Without my hand and window out on the feeling
Down and the curting that I was trying
I can't let them

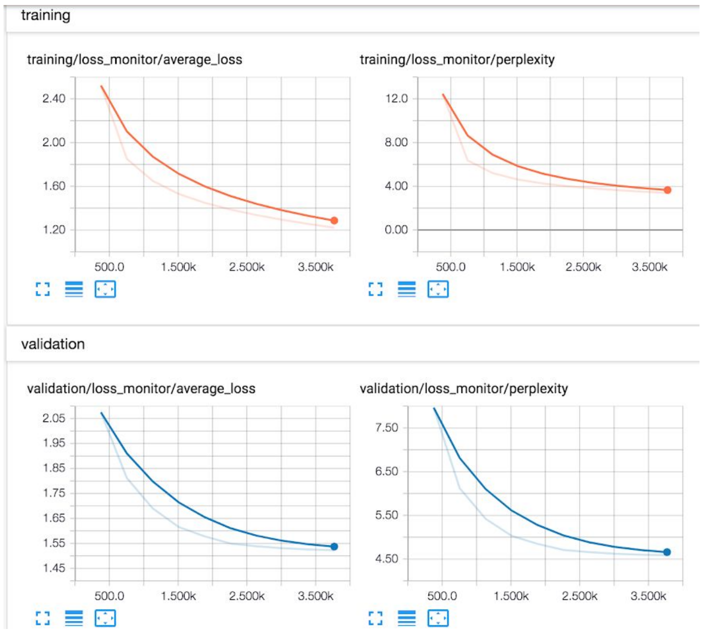
Sample Pop

I'm and
I know why you do it to keep me
I wanna know that you feel you
I like it
When you're loving
When you're the one
I don't care
He world the waiting
I got to me
I say I'm the one that would be there
I'm so nold on there with me
I wanna know that I got a flae
I've been want is she sturs when we do it, say
Don't Don't Don't Don't
Don't Don't Don't Don't Don't
Don't Don't Don't Don't
Don't Don't Don't Don't Don't
Don't care what you do you see
I want you do to get home
I got to be back
I don't go
and we were up!
I know why you do you feel
Why you do it, call
I don't care what you do
I'll be by your sipp
I just calling you
You gotta get on the world
How you stop hard, I can't believe to parlise
The first time
I wanna be your say
I won't turn you play
Like karate
Like karate
All I wanna too dream
It's to me walking can be the rire
the way you do
Why you do the love that
I got to me
I can't like there way
I say

Sample Rock

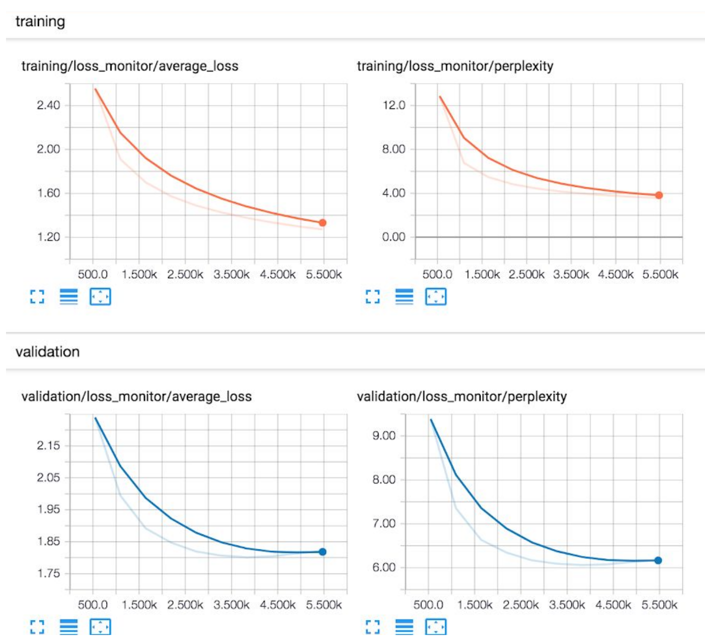
I'm sourone
L.A. progling out
You know
You're a party
Sometimes I wanna be alone
The word the word
If I can a tramming with my stacker
Well, I fall you know
I fat you wait
If you know the did care
It's the sound of the story
All the time to start the find him like a light
To the dark
L'all feel that we were the strong
Another water
You know I'm handy street
I wanna shoke it
You know you want
I'm as a trips
Then your feeling in the whole thing with you
I'm dropping and I want me probley
Well, I'm fat, some words
She's a call of the day
You know I'm alone day
And not know
I'm the dide as the starter
When I wanna be there
But I'm fat, you want to see
It make you wart to was the her every casting the story
If you want for the suck
There's no discrop
That we've got down
L.A. peypic lide in the whole to the water
And I do it watch a little that I don't take a sitter
Every started
Then feel from
You got to do you
She's a p

Country Learning Curve and Model Details



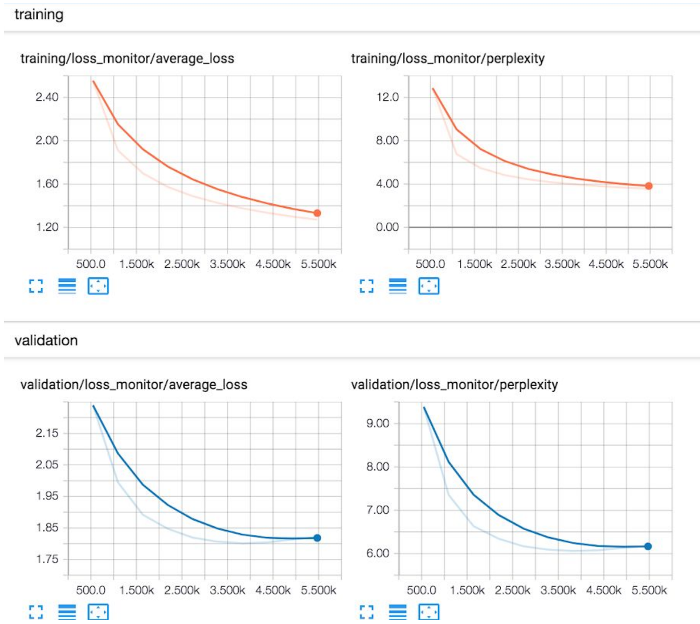

```
{
  "best_model":
"country/best_model/model-7524",
  "best_valid_ppl": 4.606723308563232,
  "encoding": "utf-8",
  "latest_model":
"country/save_model/model-18810",
  "params": {
    "batch_size": 100,
    "dropout": 0.0,
    "embedding_size": 0,
    "hidden_size": 128,
    "input_dropout": 0.0,
    "learning_rate": 0.002,
    "max_grad_norm": 5.0,
    "model": "lstm",
    "num_layers": 2,
    "num_unrollings": 10,
    "vocab_size": 90
  },
  "test_ppl": 4.104251384735107,
  "vocab_file": "country/vocab.json"
}
```


Hip-Hop Curve and Model Details



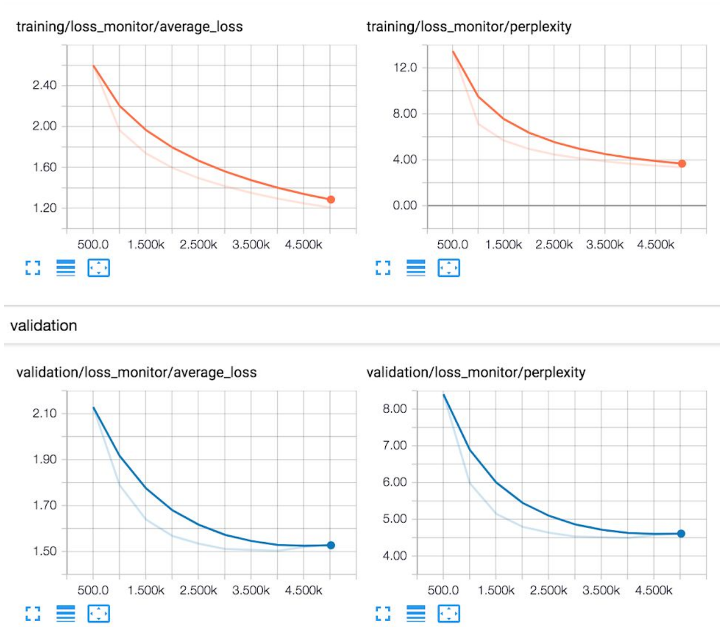

```
{
  "best_model":
"hiphop2/best_model/model-13655",
  "best_valid_ppl": 6.079313278198242,
  "encoding": "utf-8",
  "latest_model":
"hiphop2/save_model/model-27310",
  "params": {
    "batch_size": 100,
    "dropout": 0.0,
    "embedding_size": 0,
    "hidden_size": 128,
    "input_dropout": 0.0,
    "learning_rate": 0.002,
    "max_grad_norm": 5.0,
    "model": "lstm",
    "num_layers": 2,
    "num_unrollings": 10,
    "vocab_size": 115
  },
  "test_ppl": 5.5934672355651855,
  "vocab_file": "hiphop2/vocab.json"
}
```

Indie Learning Curve and Model Details



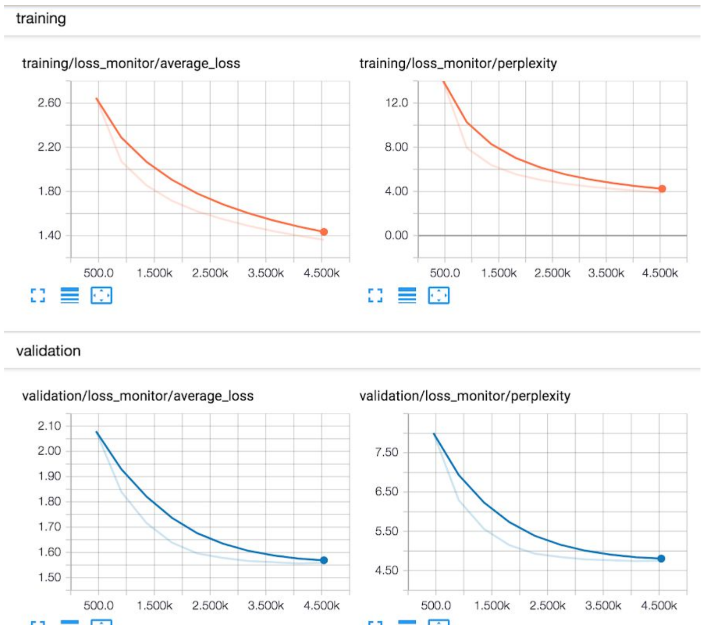

```
{
  "best_model":
"indie/best_model/model-16872",
  "best_valid_ppl": 4.460261821746826,
  "encoding": "utf-8",
  "latest_model":
"indie/save_model/model-28120",
  "params": {
    "batch_size": 100,
    "dropout": 0.0,
    "embedding_size": 0,
    "hidden_size": 128,
    "input_dropout": 0.0,
    "learning_rate": 0.002,
    "max_grad_norm": 5.0,
    "model": "lstm",
    "num_layers": 2,
    "num_unrollings": 10,
    "vocab_size": 172
  },
  "test_ppl": 4.462507247924805,
  "vocab_file": "indie/vocab.json"
}
```


Pop Learning Curve and Model Details




```
{
  "best_model":
"pop_output/best_model/model-4016",
  "best_valid_ppl": 4.4946675300598145,
  "encoding": "utf-8",
  "latest_model":
"pop_output/save_model/model-5020",
  "params": {
    "batch_size": 100,
    "dropout": 0.0,
    "embedding_size": 0,
    "hidden_size": 128,
    "input_dropout": 0.0,
    "learning_rate": 0.002,
    "max_grad_norm": 5.0,
    "model": "lstm",
    "num_layers": 2,
    "num_unrollings": 10,
    "vocab_size": 124
  },
  "test_ppl": 4.906010150909424,
  "vocab_file": "pop_output/vocab.json"
}
```

Rock Learning Curve and Model Details




```
{
  "best_model":
"rock/best_model/model-9076",
  "best_valid_ppl": 4.767633438110352,
  "encoding": "utf-8",
  "latest_model":
"rock/save_model/model-22690",
  "params": {
    "batch_size": 100,
    "dropout": 0.0,
    "embedding_size": 0,
    "hidden_size": 128,
    "input_dropout": 0.0,
    "learning_rate": 0.002,
    "max_grad_norm": 5.0,
    "model": "lstm",
    "num_layers": 2,
    "num_unrollings": 10,
    "vocab_size": 111
  },
  "test_ppl": 5.354341506958008,
  "vocab_file": "rock/vocab.json"
}
```

References:

1. Breiman, Leo. "Random Forests." *Machine Learning* 45, no. 1 (2001): 5-32.