Who's That Ghost Writer??
A Statistically Significant Score for Artist to Artist Lyric Similarity

Author
Nicholas Kim
nickimm@umich.edu

Abstract

In recent years, many rappers, most notably Drake, have come under scrutiny for ghostwriting in their songs. While these allegations have only gained real traction following the leaking of information from an insider source. Is there a way we can more formally detect these nefarious acts? In this paper, I'll present a standardized score that quantifies how unusual it is for one artist's lyrics to resemble another, by retooling the probabilities from a classification model and using bootstrapping methods to estimate the global parameters. When tested on a "synthetic" set of ghostwritten examples, it correctly labeled the potential ghostwriter 5 out of 15 times. However, the score has trouble with flagging between false positives and potentially real ghost writing cases.

1 Introduction

Drake, being one of the biggest rappers in the game, also comes with his fair share of critics and controversy. Ghostwriting being a popular critique against the Toronto-based rapper. With the most notable case emerging in 2015 with Quentin Miller, or more recently, with the now iconic clip with Soulja Boy in the Breakfast Club Podcast claiming Drake stole his "flow word for word, bar for bar!" in the release of his 2010 studio hit "Miss Me". For my project, I wanted to create a metric that will be able to spot such instances without the need of insider information. While there haven't been any attempts at approaching this problem, there have been many takes on how to parse artist specific styles using NLP. Past works, have judged similarity based on a wide array of metrics from semantics to phonetic similarities through training models with different representations of the lyrics. I built a simple bag of words model using a TF-IDF representation of the data. My innovation came in leveraging these probability scores to create a metric on lyric similarity between Artist A and B. The details of which will be fleshed out in the Methodology section of this report. When testing my scores on a set of "synthetic ghost written" examples it was able to correctly flag all but 1 of the songs to be potentially ghost written. More importantly, it correctly flagged the 3 Drake songs that were under controversy with Quentin Miller in 2015, even correctly labeling the potential ghost writer to be Miller in 2 of the 3 songs. However, after testing on an additional set of non-ghostwritten examples it falsely flagged all but 1 of the songs. Future works, will need to try to tune our model's precision to reduce the number of false positives.

In music, your voice is your commodity. Making it essential that you get the correct credit for any work you put in. Ghostwriting, or cases where an artist doesn't get credit for their work in the creation of lyrics in a song, is a direct threat against an artists greatest commodity and creative voice. Making ghost score a critical measure to keep artists from being taken advantage of.

This project has shown that you can detect lyrical similarities between artists using NLP, but fully ensuring when a song is ghost written can be a much trickier task.

2 Related Works

One of the challenges I'll be facing in this project is how to judge similarity in lyrics, as there are many factors that go into how we perceive closeness in music. Luckily, in a paper by Haven Kim and Taketo Akama, looked into different similarity metrics and there relative significance to human perception of closeness [1]. For all of the aspects tested they used cosine similarity as their quantitative measure. Topic similarity, how close is the subject matter to one another, was one of the measures they looked at (Haven Kim, 2024). To get the embeddings they used an ML algorithm that classified

[1] https://arxiv.org/html/2404.02342v1

songs into one of 50 topic classes which were then used in the cosine similarity metric (Haven Kim, 2024). Semantic similarity, how close is the context, was found using a pretrained sentence BERT model (Haven Kim, 2024). Mood was also measured, using the Deezer Mood Detection data set, where they used the euclidean distance formula between two vectors storing valence and arousal for each song in the dataset (Haven Kim, 2024). Audio similarity was also considered by creating an embedded vector of an audio file for each song then taking the cosine similarity (Haven Kim, 2024). Phonetic similarity, the pronunciation of the lyrics, was considered through an embedded vector created from first transforming the data by phonemes (Haven Kim, 2024). Lastly the measured what they called musical difference, which is a quantity of phoneme repetition degree (Haven Kim, 2024). This was found by looking at all the bi-gram combinations and dividing by the number of unique bi-grams used. To test the impact of these metrics against human perception of similarity they conducted a survey study (Haven Kim, 2024). Their findings were Semantic, Audio, and Musical Difference had the highest significance in high perceptual lyric similarity (Haven Kim, 2024). I can use these findings to help narrow down which metrics I want to use to judge for potential ghostwriting.

Instead of just using the difference in cosine similarity I could alternatively try and form this into a classification model to label a correct artist. In paper by Tunç Yılmaz1 and Tatjana Scheffler they try and do this directly using a single channel CNN model [2]. They were inspired by ground breaking discovery in text classification using a CNN compared to traditional NLP approaches with their model (Tunç Yılmaz, 2022). Here they tested 3 different models, one using character embeddings from GloVe, another using embeddings from Byte-Pari Encoding, and lastly using phoneme embeddings due to its effectiveness in NLP for poetry (Tunç Yılmaz, 2022). After training their model using CNN they were able to correctly classify correct authorship 29.8% of the time using a combination of all three models (Tunç Yılmaz, 2022). From just the base models the character embeddings performed the best with an accuracy of 23.6% (Tunç Yılmaz, 2022). What was interesting to see was that when tested in genre classification it got confused between genres that are historically linked (Tunç Yılmaz, 2022). For example, with Blues and R&B showing the model did learn some differences between genres (Tunç Yılmaz, 2022). Overall, this is a great paper that I can use as base when creating my model. If I have time, I could try implementing a CNN model and compare it to the traditional NLP to see if one performs better than another.

Since this is an NLP class the CNN method might not be the most relevant, but still offered a great alternative approach. On the other hand, a paper by Akshay Mendhakar and Mesian Tilmatine offer a look into how author classification might be done using a standard Naive Bayes [3]. In this paper they pulled data for the artists Udo Lindenberg, Konstantin Wecker, Stoppok, Ulla Meinecke, Hannes Wader, and Fettes Brot and ran a Naive Bayes model to classify authorship (Akshay Mendhakar, 2023). In addition, they ran other traditional classification models such as logistic regression (LR), support vector machines (SVM), naive Bayes (NB), decision tree classifier (DTC), K-nearest neighbor (KNN), and neural networks (NNs) to test performance against (Akshay Mendhakar, 2023). What they found was that Naive Bayes came back with the highest accuracy, with high Recall, Percsion, and F1, all around .89 to .9, showing that Naive Bayes seems to be the best model for author classification (Akshay Mendhakar, 2023). While I don't know much about these German singers. If they are from different genres of music the classification scores might be inflated as its easier to distinguish between them as they come from different musical styles. On the other hand, if they are all from the same genre of music these score are far more impressive. In either case, it shows promise that this sort of classification task is possible and I'm excited to see if I'll see similar results for my application.

The paper by Ma and others aims to tackle the challenge of modeling an artists image style [4]. Current models that have tried to tackle this issue have only allowed from one style or content input. This paper describes a multimodal approach to this problem where they used content and style descriptions of images to help with generating new images that are more representative of the reference photos artisitc style (Zhuoqi Ma, 2025). This distentagle-

[2] https://link.springer.com/article/10.1007/s42803-022-00050-x?utm_source=chatgpt.com

[3] https://jlcl.org/article/view/242/245

[4] https://arxiv.org/html/2412.14496v2

ment as they describe helps to prevent the model to only try and replicate the reference photo but instead learn the nuances of the artists style to generalize better (Zhuoqi Ma, 2025). The actual model architecture consisted of what they called a Content and Style Disentanglement Network and a Multi-Step Cross Attention Layer (Zhuoqi Ma, 2025). At a high level they used the new content and style descriptions of the image to aid in the new embeddings of the image to get a better overall generation of a new image.

A paper by Mayer and others tried to use just the musical lyrics to classify the genre of songs [5]. Very similar to other works mentioned in this section they used created new features that are specific to music. This includes a feature on the rhyme of the lyrics which they used to denote different rhyme patterns that are present in the lyrics (Rudolf Mayer, 2008). Nextly, they created a parts of speech feature wich counted the number of nouns, verbs, and etc used in a song lyric (Rudolf Mayer, 2008). Lastly, the created a test statisitc feature which was a simple count on the frequency of words used in a line or song (Rudolf Mayer, 2008). The resulting classification accuracy of their model was similar to the results I got with the highest accuracy coming from a 5 layer Neural Network with an overall accuracy of 28%.

These papers have all tackled the problem of capturing artistic style and artist similarity, but none have tried to create a metric on whether the similarity, is nefariousness or not. I'll try to quantify such a measure using some statistical techniques which could work in practice.

3 Data

For my data, I used the "lyricsgenuis" API to pull lyrics from a specified list of artists. This list included the following: Kendrick Lamar, Drake, Quentin Miller, Big L, Jay-Z, Soulja Boy, J Cole, A Tribe Called Quest, Mac Miller, Joey Badass, Tyler the Creator, and Kanye West. The API came with some limitations which influnced how I went about pulling all the songs I wanted in my corpus. The main issue was that there was no way to pull songs randomly in their entire discography without pulling all the songs individually. In order to still get a good coverage of the artists discography while keeping the pulling time manageable, I pulled data in three ways. The first of which was pulling all songs on a list of specified albums by the artist. The choice of the albums were pretty arbitrary, but I mainly emphasized picking albums that were released at different times along with including their most popular hits. For each artist this album list consisted of roughly 5-8 albums. Since lyrics are being pulled from Genuis which is a crowd sourced website not all songs have lyrics associated to a song. For artists that had a lot of songs with missing lyrics I supplemented them with more albums in this list. This was mostly for Soulja Boy, who has released a ridiculous amount of albums over the years. The second method came in pulling 50 of the artists most popular songs. This could include features/appearances in other artists that I didn't include in my corpus. Additionally, there could be overlap in their most popular hits appearing in the album data collection phase. In order, to prevent this I stored the song id in a set and if that id was unique I would then add that song to my dataset. The last phase came in pulling 50 of the artists most recent songs. Additionally this could include any appearances, including writing features, that they made in other artists songs not included in my corpus. A similar unique id check was made when pulling.

This whole data collection phase took roughly 8 hrs to pull which is the big limitation in trying to scale this up to include 100s or 1000s of other artists.

For each song, I pulled basic information like the song title and who it was released by, but more importantly, I also pulled all the artists that appeared as writer credits which would be what I would use to get my synthetic ghost written examples.

3.1 Data Preprocessing

In order for my model to learn artist specific writing styles, I broke up the original song lyrics to verses that they wrote, including adlibs. This was filtered by looking at the artist(s) inside the header of each verse using regular expression where it followed a structure like the following: "[Verse 1: Kendrick Lamar]". In cases, where more than one artist had a speaking role in the header, the secondary artist's lyrics were generally formatted inside parentheses to signify adlibs. I would then only get the lyrics inside these parenthesis for that section of the lyrics. In some cases the lyrics didn't

[5] https://books.google.com/books?hl=en&lr=&id=OHp3sRnZD-oC&oi=fnd&pg=PA337&dq=artistic+style+classification+lyrics&ots=oHPLrCgxc7&sig=2TR_dpNjOfWzLS-sI53zvNo6fmc#v=onepage&q=artistic%20style%20classification%20lyrics&f=false

include an artist name in the section headers as only one artist sang or rapped the entire song. If that was the case, I would then get all the lyrics for said song.

A lot of data cleaning was needed, especially for the 50 newest songs as there were many instances of the song not actually coming from the artist due to Genius being an open source website. These songs were then filtered out leaving us with 1,512 total songs in my dataset.

The exact number of songs for each artist can be seen in the following table:

Artist Name	Count
Drake	196
Kanye West	157
J Cole	140
Joey Badass	124
Tyler the Creator	123
Soulja Boy	122
Mac Miller	121
Jay-Z	117
Kendrick Lamar	103
Quentin Miller	100
2Pac	89
Big L	82
A Tribe Called Quest	38

Table 1: Total song count per Artist

A key bottleneck in my current project is the scarcity of data on true ghostwritten songs. This is because it's impossible to know with a hundred percent certainty that a song was, ghostwritten without the release of information from a third party. On the other hand, in incidents of highly publicized accusations of ghost writing, such as Drake and Quentin Miller, writing credit was eventually given to the wronged party.

In order to work around this issue, I used the writer credits variable to create "synthetic examples" of ghostwritten data.

Writers credit works by including all artists that had some influence on the lyrics produced in the song, either as a direct spoken feature or as background help in the writing process of the song. The idea was that while they didn't make a physical appearance in the song could our model still figure out the other artist's writing style from just the lyrics alone. To capture this idea I then created a binary variable titled "potential ghost written" returning 1 if another artist in my corpus appeared as a writer creditor but didn't have. a direct speaking feature. These songs would then be used for my test dataset as my "real" cases of ghost writing. The total number of songs after filtering came out to just 15 examples.

The remaining songs were then split into training and validation sets using a 80%,20% split resulting in sizes of 1,197 and 300, respectively.

4 Methodology

For this project I broke up my task into two phases.

1. A classification model to learn artist specific writing styles

2. A ghost score phase that used these scores to give a measure of how similar a lyric is to another artist and how unusual is it

4.1 Classification Model

To find potential ghost writers we first need to learn an artist specific writing style in hopes we'll still be able to detect which artist wrote a given lyric even if they weren't given explicit credit. These styles were learned through a Multi-class Regression model where I used different representations of the lyrics to pass into my model. I used a TF-IDF transformation as it offered a higher dimensional representation of the words over regular tokenization. For TF-IDF I used the TFIDFVectorizer from sklearn with max features of 50,000 and and n-gram range of 1 to 2. Once the lryics were transformed I then passed it into a 2-layer multi-class logistic regression model with the first layer consisting of a dimension of 128, and passed it into another layer of size 128 after passing through a ReLu. Finally, these were passed through our output layer of size 13.

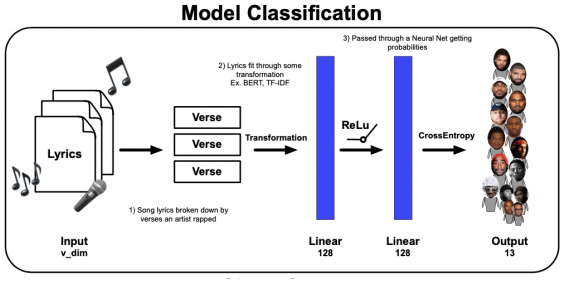

Figure 1: Visual of Classification Model

I tried two training methods, one with stochastic gradient descent and one without. For SGD, through each epoch, I would randomly select an index in my training dataset and compute its loss function using Cross Entropy, and used the AdamW optimizer with a learning rate of 1e-4. For non SGD

I would compute the loss for the entire dataset at each loop using the same loss and optimizer as before. The number of epochs was different for each method used, with SGD being trained on 15 epochs, while non-SGD was trained on 5,000. Lastly, once the model had finished training, I would test the accuracy of my model on the test dataset. The following model accuracies can be seen in the table below along with other's I tried:

Model	Val Acc	Test Acc	F1
Random Guess	20%	7%	15.16%
BERT	27.33%	20%	15.28%
TF-IDF Big w/ SGD	30.67%	26.67%	12.04%
TF-IDF	59.33%	46.67%	30.83%
TF-IDF w/ SGD	67.66%	86.67%	74.29%

Table 2: Classification Model Performance Metrics

From Table 2, we can see that the bigger models didn't end up improving our model's quality, with the smaller model using a TF-IDF transformation significantly outperforming both the Bigger and BERT models. We can visually see what our model learned in its classification using sklearns TSNE clustering algorithm to place our learned probabilities onto a 2-dimensional space.

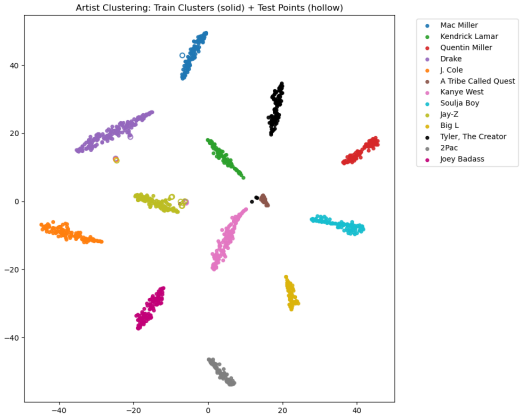

Figure 2: Clustering Plot for TF-IDF w/ SGD

In the following figure the songs used for training are seen in solid while the testing songs are shown in hollow point. We can see clearly that our model did learn an artist specific style. Of note, the bottom right region seems to be for our Old School rappers such as Big-L and 2Pac while more modern rappers generally encompass the left region. We can also see that some test songs are close to other artist clusters which is reasonable due to these songs being collaborative works with both artists. One of the songs is off of Jay-Z and Kanye's "Watch the Throne Album" and can see that its is noticeably close to the Jay-Z cluster when the lyrics were originally written by Kanye. This plot and the model's performance on the validation and test sets gives us some confidence in the probabilities outputted by our model to base our ghost score off of.

4.2 Ghost Score

Now that we have a model that can judge an artist's specific writing style we can use it to see which artist lyrics it resembles. To capture how similar a songs lyric's is to another artist I subtracted the probability given to the artist that actually wrote the song with the rest of the artists in my corpus. From there, I converted the difference scores into a probabilities with the formula $s(d_i) = \frac{1}{1+|d_i|}$. The top half of the figure below shows the steps required for our similarity score.

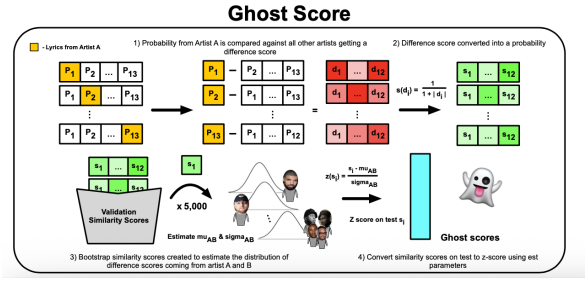

Figure 3: Ghost Score Pipeline

While the similarity score on its own is good at capturing how relate another artist style is to the actual writer, it doesn't capture whether this similarity is significant or not. Luckily, z-score or any standardized score is able to capture such a significance in a given metric. For us to then standardize our scores we need to estimate the mean and standard deviation in similarity scores when the writer is Artist A and compared to Artist B. With 13 artists in my corpus, each having to be compared to the remaining 12, this totals to us needing to estimate 156 unique distributions. To get these estimates, I then performed bootstrapping on the similarity scores produced from my validation set. This meant sampling with replacement, till I got the exact number of times Artist A came up as the original writer in my validation dataset, and computing its similarity scores with the other artists. Once that number was hit, I would take the mean and standard deviation of the similarity scores for that one sample and repeat the exact same process 5,000 times. The final estimates of my distributions would come from taking the mean and standard deviations from these

5,000 sample estimates of each parameter. I then used these estimates to standardize my similarity scores on the test set to get my final ghost score metric.

5 Evaluation

So, were we able to detect any nefarious activities? In theory, due to the songs in my test data set having been cases where another artist in my corpus had a direct role in the lyrics of the song, we should expect a relatively high ghost score for each song. However, to put these scores into perspective, I later added 14 more songs to test the ghost scores produced. All but "Miss Me" by Drake didn't have any relations to another artist in my corpus, so we'd hopefully expect low scores on these later songs. After computing these scores on my test data set, I got the following metrics, as seen in Table 3 of the Appendix.

When focusing on the original 15 songs, we can see that our model was able to get the correct ghost writer 33.33% of the time with a very strong link between Jay-Z and Kanye West. Drake and Quentin Miller were the next closest with 2 predicted instances. Our original assumption held true on these 15 songs, with all but one coming back as non-significant.

However, while these handpicked results are encouraging. When zooming out and bringing in the scores resulting from our additional 14 songs, we can see that our metric is not perfect. Even when given songs that have no connection to another artist in my corpus, we resulted in very high scores. Most notably, between Soulja Boy and Big L in his track "In My Car", and Kendrick Lamar and Quentin Miller with his tracks "Squabble Up" and "Luther".

6 Discussion

While our metric is not perfect, it is still interesting to see that it was able to detect some real instances of ghost writing. Most notably, it flagged two of the songs that were under controversy between Drake and Quentin in 2015, with "10 Bands" and "6 Man", even correctly labeling Quentin in one of the songs. This shows some evidence that our model did learn an artist's specific writing style! Other interesting examples are Izzo and Work Out. Izzo is famously known to be a track that Kanye co-wrote for Jay-Z, setting the grounds for his future success in the industry. Similarly, our model can spot such similarities in this track with the rest of Kanye's music. Work Out is a track that J Cole drew inspiration and even sampled from Kanye's own "The New Work Out Plan". While our model didn't give the correct label, it was able to spot the significance of the song in relation to other artists. The prediction of Drake is interesting as Quentin has worked with both Drake and Kanye West, which could explain some of the similarities between the two artists.

When putting our scores on the original 15 into perspective with non-ghostwritten examples, it highlights some flaws in our metric. These results highlight that our score may significantly penalize artists for drawing inspiration from another artist's style, or when they try experimenting with their flow. For example, in the two Kendrick songs recorded in my test data. These come from his most recent album of GNX, which was inspired by the boom bap era of hip hop, compared to his lyrically dense records like To Pimp A Butterfly and DAMN. Due to such stark differences in his style, it falsely flags his most recent works as not his own. These issues might be resolved with more data to train our classification model on. Another approach could be for our score to account for the time history of the musician, by comparing the scores from their most recent records in relation to those of another, compared to their entire discography.

7 Conclusion

Ghost writing is a lose-lose scenario. Either an artist doesn't get the rightful credit they deserve, or another artist's credibility is tarnished due to a wrongful allegation. My project has highlighted some of the challenges of this difficult problem. In some cases, it has shown that we can detect ghost writing, flagging songs under controversy between Drake and Quentin Miller in 2015. However, we've also shown how noisy and unreliable this metric is after testing on non-ghostwritten examples. For future work, I can look into trying to improve my model's precision by accounting for time-varying styles, as there are too many false positives being generated. This project offers a foot in the door for a potential solution to this challenging problem, which future works can use as a starting ground.

8 Other Things I Tried

As referenced in Table 2, I tried a couple of larger models in hopes of improving my model's

classification accuracy. The first of which was a BERT-style embedding instead of the TF-IDF representation. The idea was that, due to these embeddings living in a higher-dimensional space, and already including some information in its pretrained values, I could use such information to make cleaner distinctions in artist styles. However, after training my model over 50 epochs with these pretrained embeddings, it wasn't able to surpass my best-performing model. This might be due to the limitations in my data size. While a larger model is better in theory, it also requires a much richer and wider set of data to fully train to its full potential, which is something I lacked with my data set.

In a similar note, I thought maybe a larger model using TF-IDF with SGD could outperform my current model. To do this, I upped the first layer to the size of 1,000, which passed values into a 128-dimensional layer. From there, it would pass those values into a layer of size 32 and lastly onto the output layer of size 13. After training my model on 50 epochs, it again didn't outperform my smaller model, but it was a slight improvement over the BERT-style embeddings.

I also tried a simple bag-of-words method with a sparse matrix representation, like in HW 1, that performed significantly better than both of these larger models. After training, its accuracy topped off around 48 to 50% on the validation set. This led me to then try the TF-IDF representation as it offered an ngram approach, which placed greater emphasis on more meaningful words in the vocab set. Due to the model's large improvement with the TF-IDF transformation, I didn't record the simple bag-of-words model in my table figure.

9 Future Works

As mentioned, my current ghost score has a precision issue resulting in many false positives. I think this is due to the size of my data and the fact that it doesn't account for a change in an artist style over time. If I had more time, I could try to expand my dataset to include more songs and artists. Additionally, I could also try adding an era factor to my model, which would then try to learn an artist's style over different time spans. Hopefully, this would make the model learn distinct styles by an artist rather than just one overall grouping.

References

Mesian Tilmatine Akshay Mendhakar. 2023. Automatic authorshipclassification forgerman lyrics using naïve bayes. *Journal for Language Technology and Computational Linguistic*.

Taketo Akama Haven Kim. 2024. A computational analysis of lyric similarity perception. *Cornell University*.

Andreas Rauber Rudolf Mayer, Robert Neumayer. 2008. Rhyme and style features for musical genre classification by song lyrics. *ISMIR 2008: Proceedings of the 9th International Conference of Music*.

Tatjana Scheffler Tunç Yılmaz. 2022. Song authorship attribution: a lyrics and rhyme based approach. *International Journal of Digital Humanities*.

Zejun You Long Tian Xiyang Liu Zhuoqi Ma, Yixuan Zhang. 2025. Wikistyle+: A multimodal approach to content-style representation disentanglement for artistic image stylization. *arXiv*.

10 Appendix

Song Name	Artist Name	Ghost Score	Predicted Ghost Writer	Actual Ghost Writer
Hurt Feelings	Mac Miller	1.63	Drake	J Cole
Work Out	J Cole	4.73	Drake	Kanye West
Facts	Kanye West	1.18	Jay-Z	Drake
Murder to Excellence	Jay-Z	1.82	Kanye West	Kanye West
Illest Motherfucker Alive	Jay-Z	1.57	Kanye West	Kanye West
Primetime	Jay-Z	2.73	Kanye West	Kanye West
Thank You	Jay-Z	1.18	J Cole	Kanye West
Encore	Jay-Z	2.28	Drake	Kanye West
Lucifer	Jay-Z	2.43	Joey Badass	Kanye West
Takeover	Jay-Z	0.80	J Cole	Kanye West
Izzo (H.O.V.A)	Jay-Z	1.94	Kanye West	Kanye West
Legend	Drake	1.26	Jay-Z	Quentin Miller
10 Bands	Drake	2.41	Quentin Miller	Quentin Miller
6 Man	Drake	3.12	Kanye West	Quentin Miller
Pop Style	Drake	1.48	Quentin Miller	Jay-Z, Kanye West
Party on Fifth Ave.	Mac Miller	0.90	Drake	None
Smile Back	Mac Miller	2.88	Drake	None
Middle of the Ocean	Drake	1.51	Kanye West	None
Rich Flex	Drake	1.96	Kanye West	None
Miss Me	Drake	1.69	Tyler, the Creator	None
h u n g e r.o n.h i l l s i d e	J Cole	1.29	Jay-Z	None
l e t.g o.m y.h a n d	J Cole	1.45	Jay-Z	None
Squabble Up	Kendrick Lamar	4.12	Quentin Miller	None
Luther	Kendrick Lamar	4.22	Quentin Miller	None
Selah	Kanye West	1.28	Jay-Z	None
I Thought About Killing You	Kanye West	2.80	Drake	None
Only A Customer	Jay-Z	3.10	Quentin Miller	None
American Dream	Jay-Z	1.29	J Cole	None
In My Car	Soulja Boy	8.36	Big L	None

Table 3: Ghost Score on Test