Sentiment Analysis of Rapper Mentions in the GermanRap Reddit Community

s1065338
Radboud University
Netherlands

1 INTRODUCTION

For this project, I wanted to extract sentiment information from the r/GermanRap community on reddit, which I moderate. The main motivation behind this is getting factual intel that the moderation team can then base community management decisions on (such as which rappers to invite to community events) as well as to aquire a deeper understanding of the community and subculture as a whole.

Therefore, I want to answer the question which rappers are most well-regarded in the r/GermanRap community. Along with this, I have several secondary research questions, namely which rappers are the most mentioned, least well-regarded, most controversial, and how the popularity of some often mentioned rappers has changed throughout time. As the meaning of well-regarded is not quite clear, I shall explore several slightly different ways to evaluate it. To receive the results I am looking for I will first create n-grams and train a word-2-vec model on the text data extracted from reddit, which will help me identify aliases for rappers. With a dictionary of rapper names and their aliases, I shall then run a LLM-powered sentiment analysis on the text data.

2 BACKGROUND AND RELATED WORK

2.1 Word2Vec & German Language

My Word2Vec-approach is inspired by Tshitoyan et al. [10], who used it to extract latent knowledge from materials science literature. On a similar vain, I hope to find rappers and their various pseudonyms grouped together in my Word2Vec due to the context their names are used in, even though they might not be explicitly labelled as such. The idea to group synonyms in dictionaries comes from Zhong et al. [14]. For the German language aspects of my Word2Vec analysis, I rely on insights from Brito et al. [4] and Koper and Scheible [7]. While their research proves that Word2Vec works with the German language, it also shows up difficulties specific to the German language and includes findings for the best hyperparameter settings.

2.2 Ethical Considerations of using Reddit Data

As a responsible moderator of the r/GermanRap community, I try to keep in mind ethical considerations in data collection. The recent work by Norman Adams [8] on ethical Reddit data scraping helps inform my approach to using Pushshift [3], responsibly. I do not believe that my case will be as ethically ambiguous as doing medical research from user data, but it is good to keep in mind the concept of user consent and that user anonymity should always be protected. Abidin et al. [1], among others, provide examples for preprocessing and cleaning data scraped from social media.

2.3 Sentiment Analysis using LLMs

In my research on rapper sentiment analysis in r/GermanRap, I utilize knowledge from a number of papers. The work by Aggarwal et al. [2] makes me more confident in my choice of Reddit as an excellent data source for opinion mining. While they focused on using upvotes and downvotes for sentiment categorization, I want to take a different approach by implementing LLM-based sentiment analysis for more nuanced results. My planned use of LLMs for sentiment analysis is supported by recent successses in using this method. Shaikh et al. [9] have utilized a LLM model to perform sentiment analysis on student feedback and shown an impressive overall F1-score of 88%, outperforming state-of-the-art deep learning and transformer-based models. Zhang et al. [13]'s work on LLM-based sentiment analysis can provide guidance for my implementation and help design clear prompts for my scoring system to ensure consistent evaluation. Upadhye [11] also gives a comprehensive overview of the state of sentiment analysis using large language models. Lastly He et al. [6] show that LLMs outperform VADER and provide practical tips on how best to use LLMs for optimal outcomes. The model I end up using is the brand new Qwen 2.5, 3b model by Yang et al. [12]. Their model is impressive for its size and German skills, and it is the current SOTA open weights model for its size. On the other hand, the model that I want to compete against is the german-sentiment-bert model by Guhr et al. [5]. It leverages the BERT technology pretrained on a german text corpora and is the go-to for German sentiment analysis with over 200,000 monthly downloads on Huggingface. The code for this project is open-source on my GitHub: https://github.com/zitr0y/Germanrap-Sentiment-Analysis

3 METHODOLOGY

3.1 Data Collection

I used a combination of the official Reddit API and Pushshift API to scrape the needed text data. I scaped every post made in the GermanRap community since its inception and every comment within each post recursively. To get access to the Reddit API and Pushshift API, one has to go through an application procedure. Due to the limitations of the Reddit API (access to less posts) and those of the pushshift API (rate-limited), I decided to get all post-IDs via the pushshift API and scrape them one-by-one via the Reddit API. This took between one and two days.

From this json-nested post and comment data, I extracted and cleaned all sentences. I analysed the most common n-grams and filtered out texts that were bot-written, spam, regular posts, or could impact the word2vec in another way. I also cleaned the sentences

TxMM'24-'25, Radboud University, Nijmegen, Netherlands

of URLs, formatting, german-language specific 'umlauts', short sentences, and split up long sentences into subparts.

3.2 Rapper Name Detection

As stated in the proposal paper, I wanted to create a baseline for which rappers I would want to see in the Word2Vec model, so that I could verify that it works well as well as identify more rappers in their vicinity. I quickly understood that if I wanted a less biased and more complete list, I would need to get it from a more reliable source than just myself. I first tried scraping the Wikipedia German Rappers list, but it was of low quality and included more unknown than relevant rappers, so I instead started scraping a number of diverse popular Spotify playlists for names. With over a thousand names from popular as well as subgenre playlists, I had a good starting point.

This list was important for a step that I had not previously considered: Creating bi- and trigrams for the word2vec model. Here, I needed to select the right parameters of threshold and min_count so that many relevant rappers would be represented as n-grams, but not too many false positives. The test results for this can be found in the B Figure 5, with the final results being:

Ngram	min_count	threshold
bigram	4	400
trigram	3	200

With this script, I could add ngrams to the text to better represent artists with names spanning more than one word, for example Kool Savas would be represented as Kool_Savas and not as two indepentent words.

Another idea that I had to adapt since my proposal was that I initially thought I could build upon an existing German Word2Vec model. This did not make much sense, as the existing models did not include sufficient rapper names and I did want the subreddit data represented in the model. I therefore trained my own word2vec model using sentences from the scraped subreddit data.

My choice of hyperparameters (to see in Appendix B Table 3) was informed by previous research papers and projects recommending specific hyperparameters for the german language, my comparatively small (versus the entirety of wikipedia and newspaper archives) dataset and the specific goal of finding rapper names. I evaluated the performance of the model both by ensuring rappers I knew should be in it were represented in the model as well as by visually confirming that rappers of specific subgenres were grouped together (as well as other similar words being grouped together, such as English language words)

Due to the fact that I had already gotten a great list of rappers from Spotify, I used the Word2Vec only to find potential aliases for rappers existing in the database. For each rapper from the list that could be found in the word2vec, I printed the most similar words in the word2vec and manually accepted or refused each proposed alias in a custom pipeline. Luckily, and affirming the success of the Word2Vec, many aliases were grouped close to the rapper's names and could be found this way.

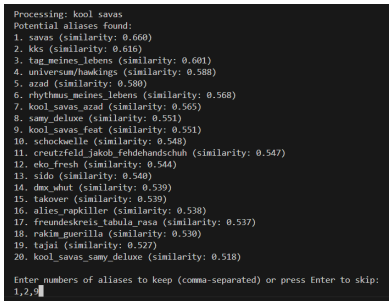

Figure 1: Pipeline for identifying aliases

Following this approach, I had a quite extensive dictionary of German rappers and their aliases, which I manually extended upon with my field knowledge. I added common typos, shorthands and nicknames to rappers and ensured they all pointed to the same canonical name.

The full word2vec can be found in the Appendix B Figure 5 or one with just rappers in Figure 4. They Are quite hard to navigate due to the sheer amounts of words, but the close vicinity of rappers of similar subgenres, rappers in general, and english words validate the resulting model (more details in the figure descriptions).

3.3 Sentiment Analysis Pipeline

Next, I was ready to tackle the sentiment analysis part. I could reuse the sentences with included ngrams from the word2vec combined with the rapper-alias dictionary. The idea was to prompt the LLM to give back a sentiment rating from '1' to '5' (very negative to very positive), or 'N/A' if the sentence did not actually refer to the matched rapper.

Finding the right LLM for the task was not easy. I created a pipeline to efficiently manually annotate a test dataset of 200 rapper-text pairs with the respective sentiment that fit the set criteria.

I then formulated a preliminary prompt that explains the task to the models with included examples and searched up tiny to medium-sized current LLM models with advertised multilingual abilities. I tested 13 models total, comparing their performance on the classification task with the annotated test dataset. My selection criteria for models were their German skills, format adherence, speed, size, sentiment detection accuracy, and accuracy in detecting if matched sentences do not refer to the rapper.

Most models did a pretty bad job in most of the criteria. Some did not adhere to the format at all (granite3.1-moe models, germanrapllm_Q8_v2), some not often enough, some were extremely slow (mixtral would have taken an estimated 55 days to evaluate the full dataset), most did not very accurately predict the right sentiment with obvious biases towards one type of rating and almost none predicted any N/A cases at all. Interestingly, qwen2.5:3b and granite3.1-dense:2b, despite being among the smallest models, had the best performance, and qwen2.5:3b the fastest speed after qwen2.5:1.5b. The full results of different models can be found in the Appendix B Table 4. (After later retesting, qwen2.5:7b now shows the best performance with the newest prompt and temperature setting and would have likely been the better model to use, and granite3.1-dense:2b in this run shows a slightly higher F1 score

as a result of random variance, but is on average slightly behind qwen2.5:3b)

Table 1: Models Evaluated

llama3.1	mixtral
germanrapllm_Q8_v2	qwen2.5:7b
qwen2.5:3b	mistral
wizardlm2	gemma:7b
aya-expanse:8b	granite3.1-dense:2b
qwen2.5:1.5b	granite3.1-moe:1b
granite3.1-moe:3b	

Next, I took the top performing small models to try and engineer the best-possible performing prompt. I went through around 20 iterations, refining prompts based on the findings. I experienced with prompts of different sizes, prompts containing examples or explanations, prompts in text and prompts in keywords, prompts with examples and instructions flipped, focus on categories or on the categorization borders, different emphasises on specific ratings and many more ideas. I found that marking the rapper in question with «rapper» significantly improved results. In the end, I hit a ceiling with a weighted F1 score of about 0.58. The final best prompt can be found in in Appendix B Table 6.

I noticed a problem while trying to find the best prompts: While granite3.1-dense:2b never predicted any N/A cases, qwen2.5:3b for most prompts around predicted a fitting amount of N/A cases, but only with around 5-20% accuracy. As most of the wrongly categorized N/A cases ended up as '3's, I made the difficult decision to scrap the 'N/A' category and just guide the LLM to a '3'-rating instead, as wrong '3'-classifications would not impact the results too much.

In the end, I decided to use a prompt with clear rules and 2-3 examples per category that clearly show the requested rating with an arrow, and use qwen2.5:3b, as it had the best performance. While it had a clear bias for '3's, so had the annotated test data as well as the whole dataset, and it did a good job at getting most categories right and rarely making mistakes that were more than one category off to the top or bottom (less than 5%).

I managed to further optimize the program by adding multi-threading and then let it run on the task of evaluating the sentiment in the around 400k texts, which took roughly 12 hours on a modern gaming PC. After running, I also managed to retroactively add time data from the original posts to the final sqlite database containing the results, which enables us to analyse temporal trends.

4 RESULTS

4.1 Quantitative Analysis

With the sentiment analysis having rated texts a total of 192,815 times of the roughly 400,000 lines of text (some of them multiple times due to multiple rapper mentions, We are left with many ratings. 73.49% of ratings are neutral, the rest is 60.91% positive and 39.09% negative with a mean of 3.038. Luckily, this distribution looks quite similar to the one of the validation set I created, apart from an over-representation of neutral cases. The distributions can be compared in the Appendix B, Figure 8 vs Figure 9.

With the obtained data linking mentions of rappers with a sentiment score and time data, we can create a number of interesting and informative tables and plots. We can see that with more mentions, the average rating will usually move closer to a neutral rating. Still, some artists such as money boy break out from this trend, as he has a high rating despite a fairly high mention count.

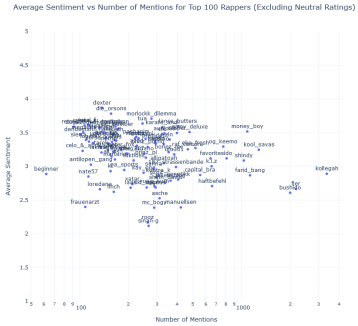

Figure 2: Sentiment vs Number of Mentions

We can also see how the top 5 rappers (for example) popularity has changed through the times. We can see that Kool Savas has been rated consistently more positively than Bushido, that Haftbefehl has become viewed less positively in the last year, and especially that the subreddit has grown considerably since 2019 and that the ratings are now more stable with more frequent mentions than before.

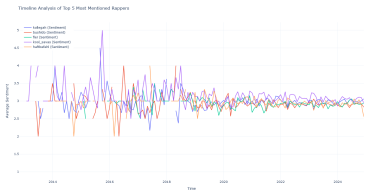

Figure 3: Timeline Analysis of Top 5

For the most well-regarded rappers, it is a bit hard to create an ultimate list. We could just sort the rappers by best sentiment scores, but that would bring rappers with very little ratings to the front due to the observed dynamics. It would be counter-intuitive that a rapper that has been mentioned once in a positive light is the most well-regarded in the community, when barely anyone knows them at all. I therefore made a number of tables: One that shows rappers with over 50 mentions sorted by sentiment score (Table 10), one that shows rappers with over 300 mentions sorted by sentiment score (Table 8), one that shows rappers with over 50 mentions sorted by sentiment score and excluding neutral ratings (Table 11) and one sorted by a composite score (Table 2) taking into account mention count, sentiment score and standard deviation. I show only the composite ranking here due to space constraints, the rest can be found in the Appendix B.

Table 2: Top 20 Popular German Rappers by Composite Score

Artist	Mentions	Avg. Sent.	Std. Dev.	Comp.
Kollegah	11800	2.969	0.636	0.581
Dexter	490	3.243	0.508	0.546
Bushido	8209	2.905	0.618	0.541
Ahzumjot	421	3.259	0.587	0.540
Amewu	382	3.246	0.563	0.540
Morlockk Dilemma	1222	3.160	0.483	0.540
Harry Quintana	365	3.263	0.599	0.539
Die Orsons	494	3.247	0.584	0.539
Fler	7832	2.907	0.642	0.536
Maeckes	425	3.214	0.553	0.536
Money Boy	2556	3.219	0.758	0.534
Lugatti & 9ine	313	3.224	0.584	0.533
9inebro	405	3.173	0.512	0.533
OG Keemo	3441	3.066	0.583	0.532
Lance Butters	1077	3.224	0.674	0.531
Kool Savas	3880	3.081	0.665	0.531
Trettmann	813	3.121	0.476	0.530
Tua	966	3.160	0.558	0.530
Samy Deluxe	1567	3.155	0.626	0.528
Karate Andi	1019	3.184	0.638	0.527

Note: Composite score combines mention frequency (popularity), average sentiment (positivity), and standard deviation (consistency) into a single metric. I weigh frequency with 0.1, sentiment with 0.7 and standard deviation with 0.2. All values are first normalized to 0-1, with higher standard deviation closer to 0 and lower closer to 1.

We can see here that even with a fairly low average sentiment and a fairly high standard deviation, Kollegah ends up on the top of this list due to his staggering mention count, even though it only gets weighed by a factor of 0.1. The rest of the list includes a mixture of smaller, well-rated and bigger, comparably less well-rated rappers. That said, all but three rappers in this list have a positive sentiment score.

I have also measured the most often mentioned rappers (Table 11), the sentiment distributions for the top 5 mentioned rappers (Table 10), the most controversial rappers (highest standard deviation) (Table 7), the most controversial popular rappers (300+ mentions) (Table 9), and the lowest rated German rappers (Table 6) in the Appendix B.

4.2 Qualitative Analysis

It is visible that rappers that produce likeable music and stay uncontroversial in their private life (Harry Quintana, Ahzumjot, 42, Goldroger, ...) show up high on the best rated rapper lists (Tables 8 and 10). On the other hand, persons that are controversial outside of rap (e.g. Reen, Sinan-G, Rooz, MC Bogy, Felix Krull, Manuellsen) as well as artists whose music is not popular with the r/GermanRap crowd (e.g. Mark Forster, T-Low, Ikkimel) are often in the lowest rated rappers list (Table 6).

In the most controversial rapper list (Table 7), we find people that have a fanbase but are also considered to produce bad music by many (Culcha Candela, Al Gear, SDP, Money Boy, Jan Delay, Zuna), or ones that are commonly considered to make good music but are controversial outside of music (Yung Hurn, Nura, MC Bogy and Reen again). In the most controversial popular German rappers (Table 9), we can see people that used to be popular but not so much anymore (MC Bogy, Favorite, Nura, Kay One, Kool Savas), ones that are very provoking (Juliensblog, 187 Strassenbande, Frauenarzt, Farid Bang, Manuellsen, Finch) and one not well regarded by oldschool rap fans (Ski Aggu).

As someone with long-term experience with this community, these results seem right on to me. The most well-regarded rappers overall are mostly underground artists with a small but committed fanbase and nothing to make them hateable, while more often occurring popular rappers are more fought over. Some are very well regarded despite their popularity (such as Money Boy), while others are past their prime and controversial or not very liked, but are very well-known (Kollegah, Bushido, Fler).

Reddit user u/Willing_Landscape_61 told me that they didn't think that LLMs were the right tool for the job of text classification and that I should try BERT models instead. Luckily, my solution outperformed the existing german-sentiment-bert model on the task when adapted to only 3 categories (see Tables 12), adding to my validation.

5 DISCUSSION

I find that the qwen-2.5:3b model did a remarkable job of sentiment analysis in German and the results align closely with the expertise I have gained in the community. The results will be helpful in providing a list of who to invite to the community in the future.

Limitations include that the GermanRap community is a separate community from the overall German Rap community, so we cannot necessarily extrapolate e.g. to which rappers are most well-regarded in Germany. There are also several cases where rappers names are too similar to common names or words, so that we cannot say for sure if the rapper was really meant or not. This is exacerbated by the fact that the LLMs proved to be bad at assessing wether a rapper was actually mentioned and I removed this.

We can also clearly see LLM biases in the selection of LLMs. We selected the LLM which, in addition to it's genuine performance, was most biased in a way that correspondent to the validation data (mostly 3's, some 4's,..). The classification looks mostly solid, but sometimes makes inexplicable mistakes. While the LLM is prompted to correctly identify rap-slang, it might not reliably do so. The time-series analysis could also be improved to be more useful.

6 CONCLUSION

Overall, we have seen that Word2Vec is a useful tool for finding aliases, that Qwen2.5:3b has impressive German skills despite its small size and Chinese origin, that LLMs can be an excellent tool for sentiment analysis, beating a popular BERT sentiment analysis model (>200k downloads a month on Huggingface) with zero-shot prompting. We have aquired valuable data about the rappers preferred by members of the GermanRap subreddit that can, in the future, inform management decisions.

I would like to see more research into using LLMs for sentiment analysis. A direct way to improve upon my results would be to use qwen-2.5:7b, which seems to perform better than qwen-2.5:3b on the final prompt I engineered. Additionally, I would like to see further improvements in BERT and LLM technology, with better efficiency and better performance despite small sizes, as inference with this model for sentiment analysis was very computationally intense.

A WORK REPORT

For this research project, I wanted to use both word2vec and sentiment analysis on the data of the GermanRap forum, as I was interested in the results I would get. This is good, as my aim to *acquire knowledge from unstructured data* was identical to the principle of text and multimedia mining. To find out more about my possibilities, I researched similar papers in the course content, with Elicit, on ArXiv and with Google Scholar. I then scribbled a visual graph overview with the structure that I imagined my project to look like. With this and my pitch, I handed in the research proposal/literature section and got valuable feedback (namely that I should be careful with how german rap might bend the meaning of words and that perhaps I misunderstood word2vec a bit, as I would necessarily have to train my own model to have my own data represented in it, instead of being able to adapt an existing one.)

I then also took my proposal to the TAs on an open lunch, and they said that it looked great and I was ready to go. My further research consisted of reading the GitHub project pages and related websites and papers of German word2vec implementations as well as sentiment analysis projects. Sadly, I found most of it to be somewhat outdated and cumbersome to prepare my data to work well with it, so I implemented everything myself, taking inspiration from hyperparameters and approaches that I encountered along the way. While chatting about LLMs on reddit, I mentioned my project and got additional input about the suitability of LLMs vs BERT models. For programming, I utilized Claude 3.5 Sonnet to be more productive while writing almost 10,000 lines of code for this project.

B ADDITIONAL FIGURES AND TABLES

Table 3: Word2Vec Model Parameters

Parameter	Value
vector_size	300
window	10
min_count	3
sg (Skip-gram)	1
hs (Hierarchical Softmax)	1
negative	15
epochs	20
alpha	0.025
min_alpha	0.0001

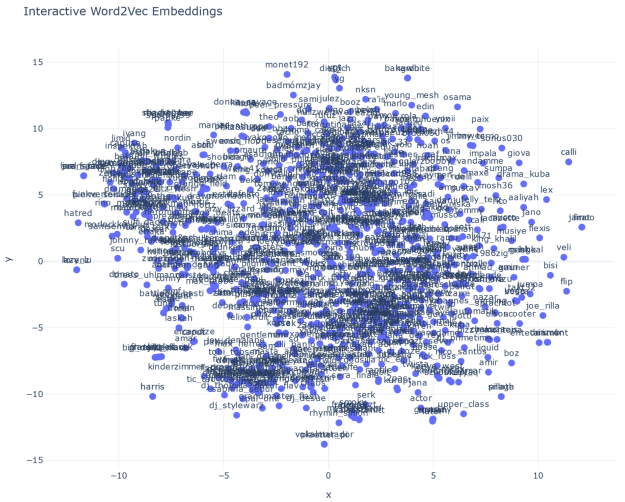

Figure 4: Word2Vec of Rappers only. We can see avantgarde-oldschoolers on the left, newer chartrappers on the bottom right, and 'allstars' near the bottom middle, old rappers in the bottom left.

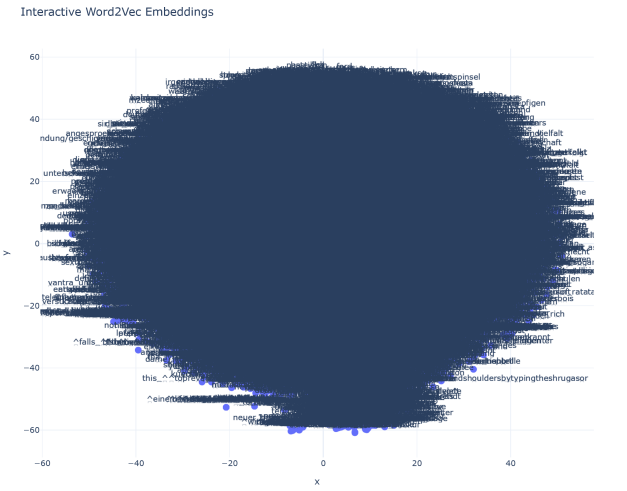

Figure 5: Word2vec all Words. English words can be found at the bottom, rappers in a small specific spot in the bottom left.

Table 4: Model Performance and Compute Time Comparison

Model	F1	Acc.	Match	Off-1	Time
qwen2.5:7b	0.643	0.640	0.640	0.295	1.4d
granite3.1-dense:2b	0.588	0.580	0.580	0.350	22h
qwen2.5:3b	0.581	0.585	0.585	0.335	19h
qwen2.5:1.5b	0.522	0.595	0.595	0.345	13h
mixtral	0.469	0.475	0.330	0.315	105d
granite3.1-moe:3b	0.407	0.503	0.500	0.455	2.0d
gemma:7b	0.347	0.306	0.300	0.465	5.2d
aya-expanse:8b	0.305	0.279	0.205	0.390	17d
mistral	0.231	0.330	0.330	0.550	1.5d
llama3.1:8b	0.224	0.345	0.285	0.44	15.5d
wizardlm2	0.208	0.290	0.235	0.455	5.2d

Note: F1 = weighted F1 score, Acc. = accuracy, Match = exact matches, Off-1 = predictions off by one level, Time = est. processing time for 400k samples (d = days, h = hours). I also measured more data like weighted disagreement, off-by-two, and errors, but didn't include them here. A difference between accuracy and exact Matches comes from some worse-performing models not being able to process all prompts. germanrapllm_Q8_v2 and granite3.1-moe:1b produced only errors. Results change slightly every run, qwen2.5:3b was generally slightly ahead of granite3.1-dense:2b.

```
instructions = """
Regeln:
- Antworte NUR mit 1, 2, 3, 4 oder 5
- Bewerte die Stärke der positiven/negativen
    Aussage:
  * 5 = sehr positiv/begeistert
  * 4 = positiv/gut
  * 3 = neutral/unklar
  * 2 = negativ/kritisch
  * 1 = sehr negativ/ablehnend
- Beachte Rap-Kontext: "krass", "brutal" = meist
    positiv gemeint
"""
examples = """
Beispiele mit Kontext:
Stark positive Bewertung (5):
"<<X>> ist der beste Rapper" -> 5 (klares Lob)
"<<X>> ist krass/brutal" -> 5 (Rap-Slang: positiv)
"<<X>> absoluter ehrenmove" -> 5 (höchste
    Anerkennung)
Positive Bewertung (4):
"<<X>> feier ich" -> 4 (klare Zustimmung)
"<<X>> macht gute Musik" -> 4 (positive Aussage)
"<<X>> hat skills" -> 4 (Anerkennung)
Neutral/Unklar (3):
"<<X>> hat neues Album" -> 3 (reine Info)
"Song mit <<X>>" -> 3 (nur Erwähnung)
"<<X>> und Y" -> 3 (Aufzählung)
Negative Bewertung (2):
"<<X>> ist nicht so meins" -> 2 (milde Kritik)
"früher war <<X>> besser" -> 2 (Qualitätsverlust)
Stark negative Bewertung (1):
"<<X>> ist müll" -> 1 (harte Ablehnung)
"<<X>> sollte aufhören" -> 1 (starke Kritik)
"""
marked_text = text.replace(found_alias.replace('_'
    , ' '),
    f"<<{found_alias.replace('_', ' ')}>>" , 1)

return f"""Bewerte das Sentiment zu {found_alias}
    im Text: {marked_text}
        {instructions}
        {examples}
        Antworte NUR mit einer Zahl von 1-5."""
```

Figure 6: Prompt template for sentiment analysis

Table 5: Performance Metrics for Different N-gram Configurations

Type	Threshold	Min Count	Precision	Recall	F1 Score
Bigram	50	2	0.56	62.80	1.11
	50	3	0.89	59.18	1.75
	50	4	1.16	53.86	2.27
	200	2	0.95	60.63	1.87
	200	3	1.53	57.00	2.98
	200	4	2.00	52.17	3.85
	400	2	1.20	58.94	2.35
	400	3	1.92	55.07	3.71
	400	4	2.51	50.72	4.78
Trigram	50	2	0.06	56.86	0.12
	50	3	0.08	49.02	0.16
	50	4	0.11	45.10	0.22
	200	2	0.08	43.14	0.16
	200	3	0.12	37.25	0.24
	200	4	0.14	31.37	0.28
	400	2	0.09	33.33	0.18
	400	3	0.12	27.45	0.24
	400	4	0.14	23.53	0.28

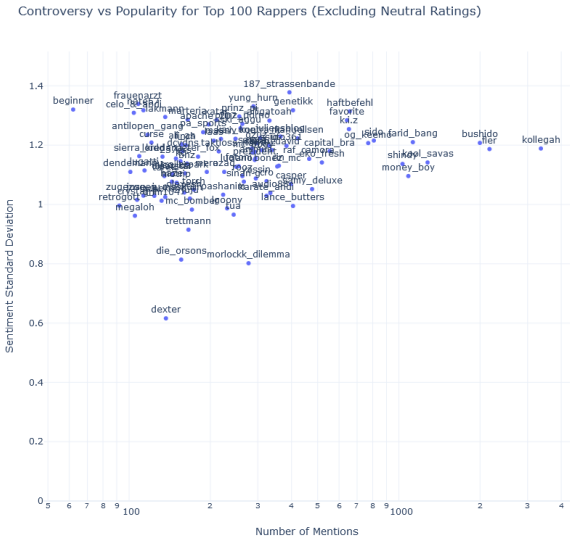

Figure 7: Controversy vs Number of Mentions

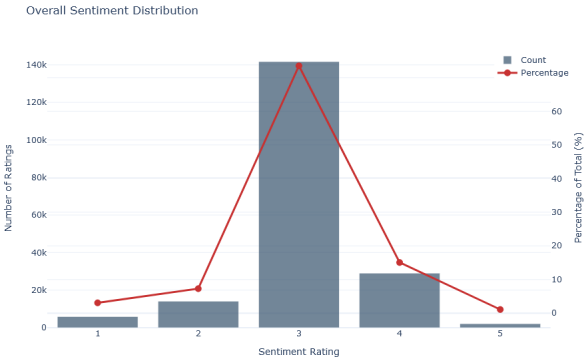

Figure 8: The sentiment distribution of the 192815 final processed sentiment ratings.

Table 6: Top 20 Lowest Rated German Rappers (min 50 mentions)

Artist	Mentions	Avg. Sentiment	Std. Dev.
Eins Zwo	205	2.434	0.526
Reen	72	2.722	0.736
Zwang	57	2.737	0.642
Sinan-G	923	2.745	0.703
T-Low	289	2.754	0.701
Kummer	283	2.760	0.709
Rooz	964	2.773	0.681
Mark Forster	290	2.779	0.706
MC Bogy	802	2.781	0.762
Cashmo	250	2.796	0.730
Mert	84	2.798	0.803
Felix Krull	61	2.803	0.703
Ego	323	2.827	0.692
Kurdo	136	2.831	0.603
Al Gear	187	2.840	0.787
Manuellsen	1610	2.841	0.680
Frauenarzt	458	2.858	0.697
Jan Delay	169	2.858	0.750
Ikkimel	127	2.858	0.601
Yin Kalle	184	2.864	0.651

Note: 'Zwang', 'Kummer' and 'Ego' are also often used in negative contexts as a German word, and 'Mert' could include mentions for other persons named Mert.

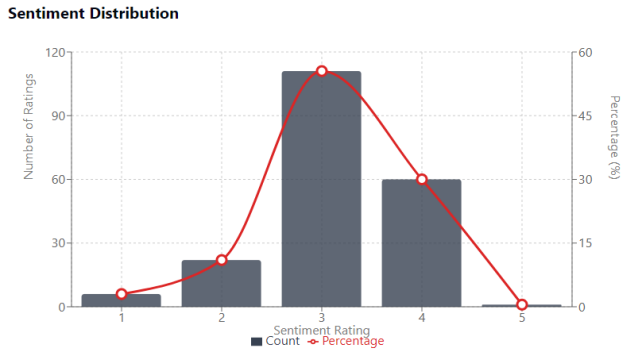

Figure 9: The human annotated ground truth sentiment distribution of the evaluation set.

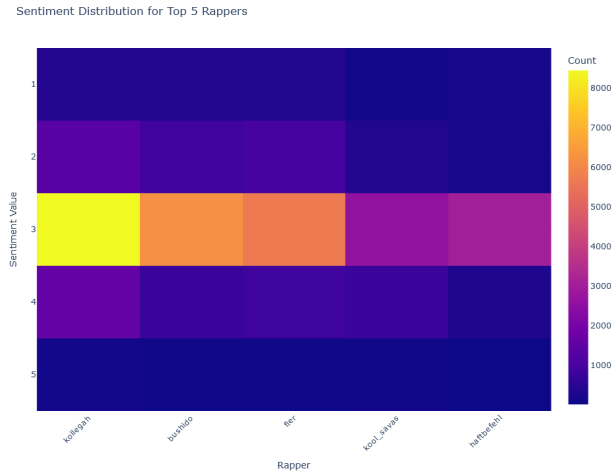

Figure 10: Sentiment Distribution Top 5

Table 7: Top 20 Most Controversial German Rappers (min 50 mentions)

Artist	Mentions	Avg. Sentiment	Std. Dev.
Omg	254	3.276	0.904
Culcha Candela	52	2.865	0.886
Nura	384	2.875	0.840
Maxim (KIZ)	282	3.259	0.805
Mert	84	2.798	0.803
Al Gear	187	2.840	0.787
SDP	87	2.966	0.784
Mosenu	62	3.177	0.779
Yung Hurn	907	2.906	0.763
MC Bogy	802	2.781	0.762
Lex	58	2.948	0.759
Money Boy	2556	3.219	0.758
Made	351	3.225	0.754
Jan Delay	169	2.858	0.750
Elo	94	3.117	0.746
Tune	115	3.157	0.744
Juri	184	3.136	0.738
Reen	72	2.722	0.736
SD	58	3.086	0.732
Zuna	119	3.008	0.731

Note: 'OMG', 'Mert', 'Tune' and 'Made' don't always necessarily refer to the rapper.

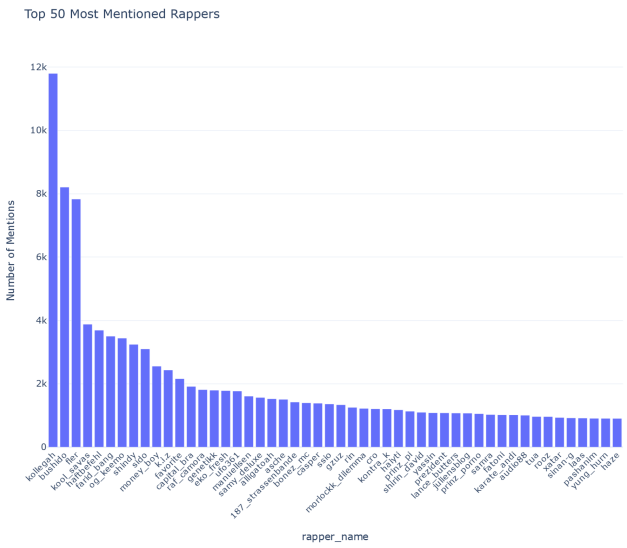

Figure 11: The 50 most mentioned Rappers

Table 8: 20 Best Rated Popular German Rappers

Artist	Mentions	Avg. Sent.	Std. Dev.	Comp.
Harry Quintana	365	3.263	0.599	0.539
Ahzumjot	421	3.259	0.587	0.540
Die Orsons	494	3.247	0.584	0.539
Amewu	382	3.246	0.563	0.540
Dexter	490	3.243	0.508	0.546
Lance Butters	1077	3.224	0.674	0.531
Lugatti & 9ine	313	3.224	0.584	0.533
Money Boy	2556	3.219	0.758	0.534
Maeckes	425	3.214	0.553	0.536
Torch	527	3.186	0.658	0.521
Karate Andi	1019	3.184	0.638	0.527
Hanybal	366	3.178	0.677	0.517
9inebro	405	3.173	0.512	0.533
Morten	432	3.162	0.571	0.525
Grim104	463	3.162	0.597	0.523
Audio88	1006	3.162	0.641	0.523
Morlockk Dilemma	1222	3.160	0.483	0.540
Tua	966	3.160	0.558	0.530
MC Bomber	600	3.157	0.580	0.525
Samy Deluxe	1567	3.155	0.626	0.528

Note: Only includes rappers with high mention counts (300+). I removed 'Made' and 'Future' from this list, as 'Made' got confused with the english word a lot and 'Future' raps in English.

Table 9: 20 Most Controversial Popular German Rappers

Artist	Mentions	Avg. Sent.	Std. Dev.	Comp.
Nura	384	2.875	0.840	0.447
Yung Hurn	907	2.906	0.763	0.465
MC Bogy	802	2.781	0.762	0.442
Money Boy	2556	3.219	0.758	0.534
Juliensblog	1072	2.928	0.723	0.474
187 Straßenbande	1423	2.997	0.723	0.489
Massiv	678	3.134	0.720	0.507
Ski Aggu	856	2.905	0.705	0.470
Sinan-G	923	2.745	0.703	0.443
Favorite	2161	3.035	0.701	0.504
Kay One	754	2.969	0.701	0.481
Frauenarzt	458	2.858	0.697	0.459
Farid Bang	3503	2.957	0.688	0.503
Animus	884	2.896	0.682	0.471
Rooz	964	2.773	0.681	0.450
Manuellsen	1610	2.841	0.680	0.468
Hanybal	366	3.178	0.677	0.517
Lance Butters	1077	3.224	0.674	0.531
Finch	555	2.890	0.669	0.469
Kool Savas	3880	3.081	0.665	0.531

Note: Sorted by standard deviation to show most controversial artists first. Only includes rappers with high mention counts (300+). I removed 'Made' and 'Ego' due to frequent confusions.

Table 10: Top 20 Best Rated German Rappers (min 50 mentions)

Artist	Mentions	Avg. Sentiment	Std. Dev.
42	238	3.592	0.593
Farhot	59	3.441	0.623
DOZ9	125	3.384	0.681
4Tune	120	3.350	0.617
Mosh36	63	3.349	0.544
Xaver	60	3.333	0.601
Battleboi Basti	100	3.330	0.637
Jumpa	82	3.329	0.649
Jace	86	3.326	0.562
Torky Tork	99	3.323	0.586
Bibiza	65	3.323	0.562
Traya	53	3.321	0.581
Shogoon	197	3.310	0.615
BRKN	117	3.308	0.725
Peat	111	3.306	0.569
Tiger104er	80	3.300	0.537
Savvy	117	3.299	0.530
Gerda	71	3.296	0.571
Gianni Suave	114	3.289	0.544
Liquid	52	3.288	0.696

Note: 'Liquid' often meant the musical genre or chemical state instead of the rapper.

Table 11: Top 20 Best Rated German Rappers (Excluding Neutral Ratings, min 50 mentions)

Artist	Mentions	Avg. Sentiment	Std. Dev.
42	155	3.910	0.502
Goldroger	77	3.883	0.648
Dexter	137	3.869	0.616
Shogoon	72	3.847	0.763
Amewu	111	3.847	0.765
Lemur	69	3.841	0.779
Lord Folter	76	3.829	0.823
Ahzumjot	134	3.813	0.796
Die Orsons	156	3.782	0.814
Maeckes	119	3.765	0.820
Harry Quintana	126	3.762	0.814
Lugatti & 9ine	92	3.761	0.869
Dietrich	74	3.757	1.004
Bazzazian	74	3.716	0.884
RAG	55	3.709	0.975
Souly	72	3.708	0.795
Morlockk Dilemma	277	3.708	0.802
Makko	64	3.703	0.810
4Tune	60	3.700	0.720
9inebro	100	3.700	0.835

Note: This list looks great to me. Small mentions could be that 'Dietrich' is part of '42', 'Maeckes' is part of 'Die Orsons' and '9inebro' is part of 'Lugatti & 9ine'. But as they are also artists on their own, I opted not to combine them as aliases. The popularity of bands as well as their members can be seen as further validation.

Model	Accuracy	Macro F1	Weighted F1
german-sentiment-bert	0.455	0.447	0.495
Qwen2.5:3b (3-class)	0.590	0.503	0.582

Agreement Type	Rate
Both Correct	0.190
BERT Only Correct	0.240
LLM Only Correct	0.337
Both Wrong	0.234
Model Agreement Rate	0.319

Class	Ground Truth	german-sentiment-bert	Qwen2.5:3b
Neutral	55.5%	29.0%	64.5%
Positive	30.5%	22.0%	12.5%
Negative	14.0%	49.0%	23.0%

Figure 12: Comparison of model performance metrics. Top: Overall performance metrics for BERT and Qwen models. Middle: Agreement analysis between models. Bottom: Class distribution comparison.

ACKNOWLEDGMENTS

I want to thank reddit user u/Willing_Landscape_61 for critiquing my choice of model architecture, which helped me connect my research better to existing research in the field. I want to thank the lecturers of the course Text and Multimedia Mining for their feedback for my project which proved invaluable, as well as the TAs of the course for their help and expertise. I want to thank Anthropic for making Claude 3.5 Sonnet, which was very helpful for doing huge amounts of python programming in little time.

REFERENCES

[1] Dodo Zaenal Abidin, Siti Nurmaini, Reza Firsandaya Malik, Jasmir, Errissya Rasywir, and Yovi Pratama. 2019. A Model of Preprocessing For Social Media Data Extraction. In *2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*. IEEE, 67–72. https://doi.org/10.1109/ICIMCIS48181.2019.8985192

[2] Archit Aggarwal, Bhavya Gola, and Tushar Sankla. 2021. Data Mining and Analysis of Reddit User Data. In *Cybernetics, Cognition and Machine Learning Applications*. Springer, 211–219. https://doi.org/10.1007/978-981-33-6691-6_24

[3] Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The Pushshift Reddit Dataset. https://doi.org/10.5281/ZENODO.3608135

[4] Eduardo Brito, Rafet Sifa, Kostadin Cvejoski, César Ojeda, and Christian Bauckhage. 2017. Towards German Word Embeddings: A Use Case with Predictive Sentiment Analysis. In *Data Science – Analytics and Applications*. Springer, 59–62. https://doi.org/10.1007/978-3-658-19287-7_8

[5] Oliver Guhr, Anne-Kathrin Schumann, Frank Bahrmann, and Hans Joachim Böhme. 2020. Training a Broad-Coverage German Sentiment Classification Model for Dialog Systems. In *LREC 2020*. 1627–1632.

[6] Lu He, Samaneh Omranian, Susan McRoy, and Kai Zheng. 2024. Using Large Language Models for Sentiment Analysis of Health-Related Social Media Data: Empirical Evaluation and Practical Tips. (2024). https://doi.org/10.1101/2024.03.19.24304544

[7] Maximilian Koper and Christian Scheible. 2015. Multilingual Reliability and Semantic Structure of Continuous Word Spaces. In *Proceedings of the 11th International Conference on Computational Semantics*. 40–45.

[8] Nicholas Norman Adams. 2024. 'Scraping' Reddit Posts for Academic Research? Addressing Some Blurred Lines of Consent in Growing Internet-Based Research Trend during the Time of Covid-19. *International Journal of Social Research Methodology* 27, 1 (2024), 47–62. https://doi.org/10.1080/13645579.2022.2111816

[9] Sarang Shaikh, Sher Muhammad Daudpota, Sule Yildirim Yayilgan, and Sindhu Sindhu. 2023. Exploring the Potential of Large-Language Models (LLMs) for Student Feedback Sentiment Analysis. In *2023 International Conference on Frontiers of Information Technology (FIT)*. IEEE, 214–219. https://doi.org/10.1109/FIT60620.2023.00047

[10] Vahe Tshitoyan, John Dagdelen, Leigh Weston, Alexander Dunn, Ziqin Rong, Olga Kononova, Kristin A Persson, Gerbrand Ceder, and Anubhav Jain. 2019. Unsupervised Word Embeddings Capture Latent Knowledge from Materials Science Literature. *Nature* 571, 7763 (2019), 95–98. https://doi.org/10.1038/s41586-019-1335-8

[11] Akshata Upadhye. 2024. Sentiment Analysis Using Large Language Models: Methodologies, Applications, and Challenges. *International Journal of Computer Applications* 186, 20 (2024), 30–34. https://doi.org/10.5120/ijca2024923625

[12] An Yang et al. 2025. Qwen2.5 Technical Report. *arXiv* (2025). https://doi.org/10.48550/arXiv.2412.15115

[13] Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023. Sentiment Analysis in the Era of Large Language Models: A Reality Check. *arXiv* (2023). https://doi.org/10.48550/ARXIV.2305.15005

[14] Ying Zhong, Valentin Thouzeau, and Nicolas Baumard. 2023. The Evolution of Romantic Love in Chinese Fiction in the Very Long Run (618 - 2022): A Quantitative Approach. In *CHR 2023: Computational Humanities Research Conference*. https://ceur-ws.org/Vol-3558/paper193.pdf