**Finding Concerts using Large Language Models : The Stockholm Concert Database as Case Study for ’Touringbot’**

**Brandon Farnsworth**

**Introduction**

This project came out of a curiosity that slowly grew into what I think is a new way of studying musical concerts at a distance.

What I want to present today are the preliminary results of a project that grew pretty organically out of playing around with some new digital history methods, and how they could be used to make new kinds of questions possible in the study of music history.

When I first started experimenting with ChatGPT, I wondered, as probably many here have, how I could use this new tool in my research. At the time, I was working a lot on social network analysis, and getting interested in identifying patterns in how concert programming is changing across Europe. Specifically I was interested in investigating how phenomena like gender mainstreaming and an increased focus on sustainability are changing how musicians perform, for example by working more nationally.

I had started studying this by manually entering concert data into a spreadsheet, which obviously took ages. Realising that Large Language Models, or LLMs, could be useful for potentially automating this task, I began experimenting with the feasability of using them to help me with data entry and structuring information about concerts.

While I could expand at length on why where artists perform is highly interesting for new research agendas in my field and is already showing promising results, I want to use the opportunity of this digital history conference to present more about the method, as well as a historical example that I’ve used. *The goal here is also to emphasise the wider applicability of this method to all manner of historical events, like protests, political rallies, etc.*

With this being said, the **problem we are going to address today** is : **how can we we use digital history methods and large language models to help extract structured information about concerts?**

In order to explore this question, I’m going to focus on a methodological validation study I did, which developed an approach to studying historical data from Stockholm newspapers and comparing it with a dataset generated by a colleague.

The data focuses on three newspapers, Aftonbladet, Dagligt Allehanda, and Nya Dagliga Alehanda, investigating the years 1848, 1858, and 1868.

These years were chosen because they correspond to an existing dataset called the Stockholm Concert Database, where my colleage Ann Reese Wilen, now at University of Gothenburg has gone through Stockholm newspapers in ten year slices between 1848 - 1908 and created a database of concerts. This presented an ideal case for testing the method: a fully-human data entry could be compared to the automated LLM results. I was further limited to just the years 1848, 1858, 1868, as these are over 150 years ago and therefore available without restrictions from the Kungliga Bibliotheket.

So we have the newspaper data on the one hand, and data made by a human on the other. The question became: *could I use a LLM to generate data that was as good or better than the human?* Larger questions about using these results for example for Social Network Analysis will have to be explored in another paper.

~~The project is based on the premise of collecting and analysing trace data, assembling these ‘records of activity’ (Keusch and Kreuter 2021, 100) together to create new understandings of social life in the past. As we are dealing here mainly with advertisements and concert reviews from newspapers that have been scanned and OCR’ed, we are working with digit~~*~~ised~~* ~~data that carries additional considerations.~~

**Method**

The data extraction method was developed at the same time as my digital humanities and programming skills, and can broadly be divided into three processing steps: 1. Collecting 2. Processing 3. Analysing.

For the purposes of this short paper, I’m going to follow just one example through the data colleciton process, which looks like this: [show slide]

1. Collecting

This involved downloading materials using the KB API endpoint that they have recently created, limiting based on the newspaper names and search terms in the given years. Because content was not reliably divided into discrete ‘articles’ by their system, I found it was better to save large ContentBlocks around the search terms to make sure I got all the info and it was not cut off.

Search terms were generated very generously: I searched for broad keywords like Konsert and Musik, as well as for the names of venues extracted from the the Stockholm Concert Database. The goal here was to get as much material as possible while limiting the amount of credits that I would need to pay for when doing the final LLM processing which would do the actual combing through the dataset.

I needed to get special access to the KB API endpoint to be able to download large enough amounts of data without getting rate limited. I did this by contacting KB, who granted me a one-week access token at a higher limit.

After downloading the data, I was left with a large database of raw OCR’ed data downloaded from KB that were ready to be combed through by the LLM.

**2. Processing the data using the LLM:**

Once I had this data, the processing itself was maybe the most straightforward. I developed a system prompt explaining the task to the LLM, attached the content to be analysed as user input, then asked it to return a JSON object containing matches for any concert events that it found.

GPT 4o mini was chosen mainly for its low cost (vs. 4o), but also because of its structured output feature, and because it is readily accessible in terms of access to tokens/second.

The structured output feature is a mode that can be toggled in the API which guarantees the model always returns a valid JSON object in the schema specified. This was useful for specifying the chain of thought structure.

The responses are then extracted and put into a SQL database of concert events.

The **prompting** itself used some basic techniques:

1. Basic Retrieval Augmented Generation (or RAG): telling it the date of the newspaper from the metadata. This was because I found that it was able to use this information to help it think through to finding the right date when ads spoke about “concert tomorrow” or “next sunday”.

2. giving it a one-shot example of what the process of going from newspaper text to structured JSON. This helped direct the LLM to better understand the task at hand.

3. Implementing chain of thought using structured output. This was in response to finding that the original OCR’ed data from KB is quite messy, leading to the LLM having major difficulties in the beginning. The chain of thought meant that the LLM had to think through trying to rewrite the patchy OCR’ed text before trying to find the info it was looking for, which ended up being very helpful for both the LLM and for myself to understand where it was struggling. You’ll notice too in the chain of thought that it’s been instructed to put everything in English. This was as I had found it tends to get less confused when it’s working in just one language, even if the inputs are in a different language.

Here is an example of what this chain of thought looked like. I’m just including a few of the interesting steps where you see the reasoning most clearly:

**Example 1:**

This shows chain of thought being used to find and confirm the venue, also using some of its own training data about the context. Notice also the use of a confidence level. Measuring how accurate this is is another area I want to investigate further.

**Example 2:**

Same thing again, using its training data

**Example 3:**

And here we see its flexibility, able to normalise names direectly in the data entry process, greatly simplifying later work.

**3. Analysing the Data / Results**

Now that I had my database full of events, the next step was to figure out how accurate the extraction was. Here I ran into a problem that the human data for these years in fact didn’t have very good date accuracy.

I struggled for a long time with being able to find the right ways of analysing the data, I settled for this conference on a multi-level matching approach. It might have been a bit overengineered for this but it seems to give a good impression that aligns with the opportunity samples that I did comparing the two datasets as well as comparing the LLM data to the newspapers themselves.

The script uses a weighted similarity approach that considers three key aspects of each event: the venue, the event name, and the event type, using a fuzzy matching on each column and finding the best match.

Weights: {‘venue': 0.6, 'name': 0.3, 'type': 0.1}

This leads to the following result:

Similarity Score Statistics:

**Mean: 80.20**

**Median: 78.90**

25th Percentile: 73.20

75th Percentile: 89.40

We can see from the distribution of scores that *the system clearly works*, as evidenced by the high amount of strong matches, but that there is a significant tail of worse matches. It’s also important to note that for this first version, I didn’t test how well the LLM was able to match the programmes and artists, just the venue names, dates, and concert titles, which can sometimes be a bit of a subjective choice, based on spot checks of both kinds of data.

Another aspect I was not able to investigate further was whether working with larger models than the estimated 8 billion parameters of GPT 4o mini, as these would likely perform much better on the reasoning tasks. This was mainly because of the higher cost of working with 4o or other larger models. I also have not yet experimented with training a model, but think this would likely be the largest performance boost. Dynamically caching responses listed as high confidence, like is possible with Anthropic’s models, would also have been an interesting road to explore.

Furthermore, the comparison approach has limitations though and is not perfect: Ann Willen, who did the human data entry I’m comparing against, made decisions about what kinds of concerts she wanted to enter, focussing on results that seemed high-quality to her. We can see this discrepancy by looking at the number of events found by the LLM vs the Human over time:

This clearly demonstrates that further investigation is needed to determine the quality of the LLM’s extra matches, in particular the spike in 1868 which may still be a bug from deduplification

However that it finds more matches than the human is itself a positive sign, as the human data entry may also have missed concerts. I ran into at least one or two cases where the LLM actually found concerts in the newspapers that were not in the human data. In one case, this was because the concert information was extracted from a review where the factual details were scattered throughout the review text. The LLM was able to reconstruct this, but presumably the human missed it as the concert was not in their database.

**Discussion and future research**

As I have already mentioned, this effort has been my first foray into more quantitative research methods, having until now mainly worked with qualitative data. However the results have been extremely impressive and have gestured at a major new research agenda in musicology analysing concerts using this ‘distant reading’ technique.

I have already begun applying and refining the same methods to the study of contemporary concerts as part of a spin-off company from the university that’s showing lots of interest from music institutions.

Working with contemporary concerts has the advantage of working with born-digital sources, meaning no worrying about OCR errors. More importantly though, preliminary results gathered by looking at the reasoning steps show that the LLM is able to draw more from its training data, which seems to have more information from contemporary sources, though this would need to be proven. The implications however are quite major, as they allow us for the first time to analyse live musical performances like we are able to with digital music.

**References**

Keusch, Florian, and Frauke Kreuter. 2021. ‘Digital Trace Data’. In *Handbook of Computational Social Science, Volume 1*, by Uwe Engel, Anabel Quan-Haase, Sunny Xun Liu, and Lars Lyberg, 1st ed., 100–118. London: Routledge. <https://doi.org/10.4324/9781003024583-8>.

Reese Willen, Anne. 2023. ‘Wilhelm Stenhammar as an Actor in the Late 18th Century Musical Life of Stockholm: Suggestions for New Perspectives and Interpretations through Digital Musicology’. *Svensk Tidskrift För Musikforskning / Swedish Journal of Music Research*, December. <https://doi.org/10.58698/stm-sjm.v105.13639>.

**Biografi**

Brandon Farnsworth is a researcher in musicology and music curator based in Berlin and Malmö. In 2024, he completed a Swiss National Science Foundation Postdoc at Lund University, Sweden examining the effects of diversity policy on experimental music using ethnographic methods. In 2020, he completed his PhD in Dresden with the publication *Curating Contemporary Music Festivals* (Transcript). Brandon has worked with, among others, Malmö Konsthall, Ultima Festival Oslo, Montreal New Musics Festival, Sonic Matter Zurich, DAAD, and the Berlin New Music Society.