What a Nano-GPT can (not) tell us about Spoken Language

Michael T.L. Pace-Sigge[a][1]

[a] University of Eastern Finland

Abstract

After the launch of Chat-GPT in autumn 2022, a lot of research has focussed on the quality and near-naturalness which Large Language Model (LLM)-based tools present in the texts produced. While one area of research focussed on the similarities and differences between machine-produced and human-produced output (e.g. Berber Sardinha, 2024), others explored in how far such tools could process more complex tasks (e.g. Valmeekam, et al., 2023; Curry et al., 2024.

While it can be assumed that Chat-GPT makes use of written-to-be-spoken training material, there has been no investigation, as yet that looks at in how far a Generative Pre-trained Transformer (GPT) algorithm is able to process (transcribed) natural, colloquial language. This research will investigate whether spoken language transcripts lead to processing difficulties; whether such generated language can be seen as a suitable reflection of natural speech; and whether machine produced texts offer new insights into the workings of language.

Keywords

Generative Pre-trained Transformer (GPT), Large Language Model, nano-GPT, Scouse, spoken English, word clusters

1. Introduction

This investigation will look at one area in which publicly available tools (being based on written material) does not cover, namely, in how far a Generative Pre-trained Transformer (GPT) algorithm is able to process training material that is based on transcripts of casual conversations. Following on from that, this paper will show in how far GPT-produced text mirrors or diverges from results obtained using a standard concordancer (WordSmith Tools 8.0) working with the same material of transcripts of spoken exchanges.

Corpus-based work provides frequency lists of the words found in the source material and enables a researcher to find highly frequent clusters of words and typical collocations for nodes. By calculating keywords, the characteristic use of words and sets of words in one corpus can be directly compared to another. While a relatively small corpus can yield crucial insights, using such a very small corpus as the training data for a Generative Transformer does have inherent problems which means that the text produced lacks natural fluency. For example, Wang et al 2021) point out that "NLG models trained from small training data still has a certain chance to generate low-quality samples". Crucially, the rather unstructured source text material shows up limitations of the algorithm when directly compared with written-text material. On the other hand, salient word nestings – their collocates and how they chunk into larger, meaningful units – also appear in the machine-produced texts and could, therefore, be seen as an alternative route to pinpointing corpus characteristics.

There are marked differences between the wording and structuring of casual spoken utterances and written English. This makes spoken language a fertile ground for further linguistic investigation. When it comes to corpus linguistics and, indeed, training data for machine learning (ML) algorithms, spoken language is put at a clear disadvantage compared to written material – namely that the spoken word

[1] *Digital Dreams and Practices, March 05-07, 2025, Tartu, Estonia*

EMAIL: michp@uef.fi (A. 1)
ORCID: 0000-0002-5164-5242

needs to be collected, transcribed and made ready to be machine readable first – a difficult and time-consuming process compared to the relative ease with which trillions of written words can be hoovered up[2]. As a consequence, the focus of this research uses what would be deemed to be a minute amount of training data – just over a quarter million words – to see whether a generated version of spoken English can be seen as a way to either observe in how far corpus-based research findings of the original text is being matched or whether this method allows researchers to pinpoint further characteristics salient of the original corpus (that is, the training material). Furthermore, in order to see whether the findings are particular to the set-up of this investigation, a parallel check with prose texts as training materials will also be undertaken.

This particular investigation wants to test three hypotheses:

1. that spoken language is too unruly and that a large number of individual speakers are too idiosyncratic; furthermore that using a variety of sources make the training material sup-par in quality compared to and equivalent written-text corpus
2. LLMs can be a reflection of language use based on natural occurring speech as training
3. Using a Transformer-generated text can reveal details of language use beyond what traditional corpus linguistics methods can do.

2. Recent related Literature

Søgaard gives a brief introduction how GPT models work. As it is a useful introduction how Transformers work:

> The Transformer architecture injects an inductive bias by segmenting the input into words and morphemes. This inductive bias is a linguistically motivated bias: While learned tokenizations do not always align perfectly with how linguists would break a text into meaningful units, it correlates much better with a gold-standard segmentation than a random segmentation would. (Søgaard, 2022)

A key take-away message from this is that the algorithm is fully disinterested what the tokens represent. In other words: input governs output to a very large degree. This does mean, as well, that low-quality data or data with a high degree of noise will be reflected in any generated output. There are, however, a number of critical voices which see concrete limitation to the current stage of Large Language Model developments. For example, Dohmatob and colleagues 2024) posit that the hunger for ever-larger data sets by tool developers might eventually lead to an implosion of the whole project. Similarly, even current approaches to computer generated texts, even when highly curated, tend to show a shrunken linguistic range and variety as Pace-Sigge and Sumakul 2022) or Guo et al. 2023) have demonstrated. A counter narrative would be to develop a technology which is able to turn small but highly curated sets of material into the key source for computational text generation.

When looking at the research at hand, earlier corpus linguistic work has already shown that the lexical choices and grammatical patterns of *spoken* English differ concretely from written English. That just a small number of distinguishing features allows to claim that "text is written rather than spoken" was highlighted as early as in the OSTI report by Sinclair, Jones and Daley (1970) [2004] and has been described in detail variously (amongst others, Carter 2004), O'Keefe, McCarthy and Carter 2007) or Cheng 2012)).

Furthermore, research by Wray 2002) has shown how a lot of spoken language seems to rely on prefabricated chunks leading also to a higher rate of repetition than found in (often highly edited) written material. Biber et al. 2000) highlight that the use of *deictics* and *vagueness markers* is also characteristic of spoken language. Sardinha 2024) looked at the human-like quality of Chat-GPT produced text, "on the assumption that, in order to evaluate the quality of AI-generated texts, one must consider register. Previous corpus-based analysis of human-authored texts has provided ample evidence that register is a

[2] 2024 has, compared to 2004, the huge advantage that we have tools – some of which provided by OpenAI – which drastically reduce the time it takes to transcribe from audio. However, the crucial issue of data collection remains. Academia is here at a cruel disadvantage as collections of spoken material will aim for a carefully weighted mix of speakers who will have to give permission for data to be used. Contrast this with tech companies who have potentially access to huge troves of spoken material (a fact highlighted by Robbie Love in a personal conversation).

key predictor of linguistic variation". Crucially, for this research, Sardinha includes a section on 1,000-word exchange between two speakers generated by Chat-GPT, which he then compares to BNC-2014 natural occurring data. Based on Sardinha's model, Chat-GPT was tasked to generate an exchange between two Liverpool speakers (see Appendix A). While nothing is known about the training data employed by OpenAI, there are a number of curiosities in the text, such as initial *h*-dropping and final *g*-dropping which is not necessarily typical only of Liverpool English, but might be seen as a marker of colloquial speech. Furthermore, there is accent variation ("bin" for "been")[3]. Crucially, the short text has a far higher occurrence than can be expected in natural use of local terms ("rammed", "scran") – almost none of which appear in the SCO corpus used for this paper itself. If anything, Chat-GPT reveals a kind of hyper-priming and over-stereotyping which results in something that (unlike Larkin's play) appears like a very bad script for a play. While it looks like ChatGPT being able to comply with the prompt, the training data, at best, appears to be written-to-be-spoken texts and reference works, but not transcripts of human casual conversations. Yet a focus on spoken language should be seen as an important step when developing artificial general intelligence (AGI). Landgrebe and Smith 2023) argue that mastery of language is both "a necessary and a sufficient condition for AGI". They propose an enhanced, more direct version of the Turing test for which one of the key conditions is "that the conversation would be in spoken form"[4]. This demand can be seen to encourage research into spoken language to form a fundamental step when developing proficient A.I. tools.

3. Methodology

For the research at hand, the idea is to use a very small data set to check what a GPT would do with transcripts of spoken English. Therefore, the training set would be smaller than 1,000,000 words – in fact, for this research the training material amounts to just below 300,000 words. To be precise, the material was originally collected to create a corpus of *Liverpool English*. The rationale behind this is that the spoken form is specific to a particular speech community and also represents material which is not published on the internet[5]. The training material consists of the SCO corpus (Pace-Sigge, 2013), transcripts of conversations recorded by two academic colleagues (Amanda Cardoso and Marten Juskan), short transcripts from BBC radio Merseyside as well as the transcripts of conversations of Liverpool residents held in the BNC Spoken 2014 (Love et al., 2017).

Karpathy 2024), who developed nano-GPT, used the works of Shakespeare as his demonstration corpus. This is by all accounts a very small data-set to train the GPT – the total number of tokens is just below one million words. Yet even this highlighted, despite producing a large number of non-words, that their nano-GPT can produce a semblance of Shakespearean drama as its output.
Initially, it was hoped that, with the assistance of Google's BARD (in July 2023), even a lay person can set up a small GPT: According to BARD, it is possible to "build your own nano GPT by following the instructions in the nanoGPT GitHub repository. For this, one needs to install Python 3, NumPy, PyTorch, and the transformers library. Once these dependencies have been installed, the train.py script can be run to train one's own nano GPT model".
However, I myself was not able to make this work beyond the first three steps. Instead, my colleague Evangs Mailoa created the BabyGPT in *Google Colab* and, after a trial run using the complete works of Shakespeare (as shown in Bhatia, 2023), the set-up for my own data set was ready. Google Colab has the advantage that it is fairly easy to set-up and use. Furthermore, up to a certain degree of usage, there is no charge for employing it, which made initial trial runs cost free. In order to train the nanoGPT and generate sufficient amount of texts, a paid-for subscription was used. Once set up, the first step needed was to turn the multifile-corpus which is the training material has to be bundled into a single .txt file.

[3] It might be that OpenAI bases this on Jonathan Larkin's play "Paradise Bound", set in L8 and published 2006 which might have been the training data.
[4] In line with many other critics, they highlight the obvious limitations – that a machine has no physical and visual context, lack human-like interpretation skills, etc. which make them conclude that AGI cannot ever attain human-level comprehensive intelligence. It must be noted that Søgaard (2022) is rather critical of their approach.
[5] This way, none of the material investigated here would have been used as part of the training material OpenAI, Gemini etc obtained by web-scraping.

This file was deposited on the Google Drive and the drive mounted in Colab and then the *read* command will be enacted.

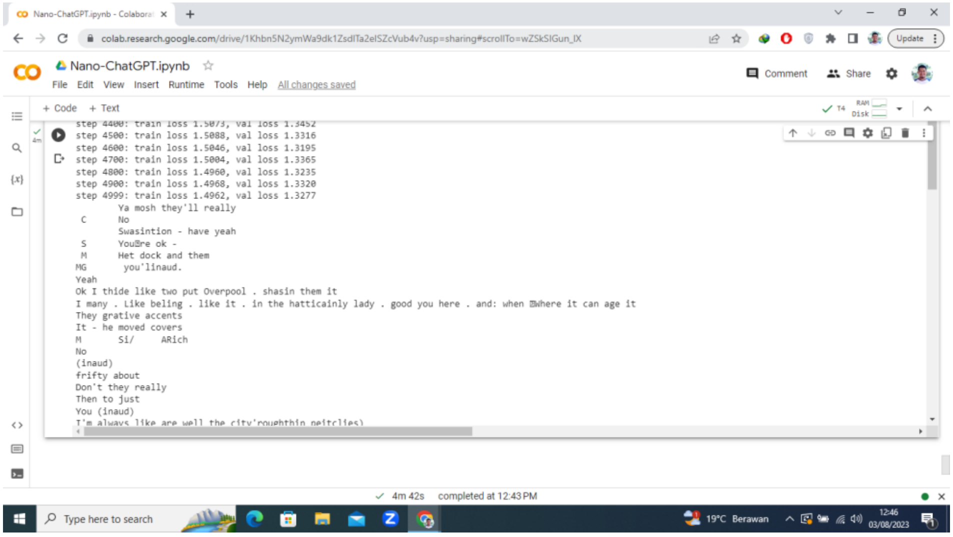


Figure 1. Initial results – a BabyGPT trained on the SCO file generates text

Once the programme was set up in *Google Colab*, the training and output settings continued to be tweaked. Thus, initially, in order to expand the training cycle to 30.000 iterations in order to tune *train* and *val loss*. However, running so many iterations does come with little gain in quality though it requires far more computing time and resources. A change (tuning) of the hyperparameters on the top of the code was needed – focussing on *n head* and *embd.*

```
# hyperparameters
batch_size = 16 #banyaknya data sekuens yang akan diproses secara
paralel
block_size = 32 #maksimum banyaknya data untuk prediksi
max_iters = 30000 #banyaknya perulangan/iterasi
eval_interval = 1000
learning_rate = 1e-3
device = 'cuda' if torch.cuda.is_available() else 'cpu'
eval_iters = 200
n_embd = 64
n_head = 4
n_layer = 4
dropout = 0.0
# ------------
```

Figure 2. First set of hyperparameter fine-tuning

Further fine-tuning of the programme in *Colab* was provided by my colleague Vishwanath Singh, who tweaked the parameters within a range that aimed to obtain the best results for the small training corpus available (see Figure 1).

```
# hyperparameters
batch_size = 128 #banyaknya data sekuens yang akan diproses secara
paralel
block_size = 186 #maksimum banyaknya data untuk prediksi
max_iters = 12433 #banyaknya perulangan/iterasi
eval_interval = 450
learning_rate = 1e-4
device = 'cuda' if torch.cuda.is_available() else 'cpu'
eval_iters = 1000
n_embd = 320
n_head = 8
n_layer = 4
dropout = 0.2
```

Figure 3. Final hyperparameters used

The revision and fine-tuning led to several test runs until the results seemed to reach a certain plateau with regards to the textual quality. Once this point was reached, four sets of generated texts were created to form the comparator material.

Table 1.
SCO corpus and ‘fake’, generated Liverpool spoken exchanges

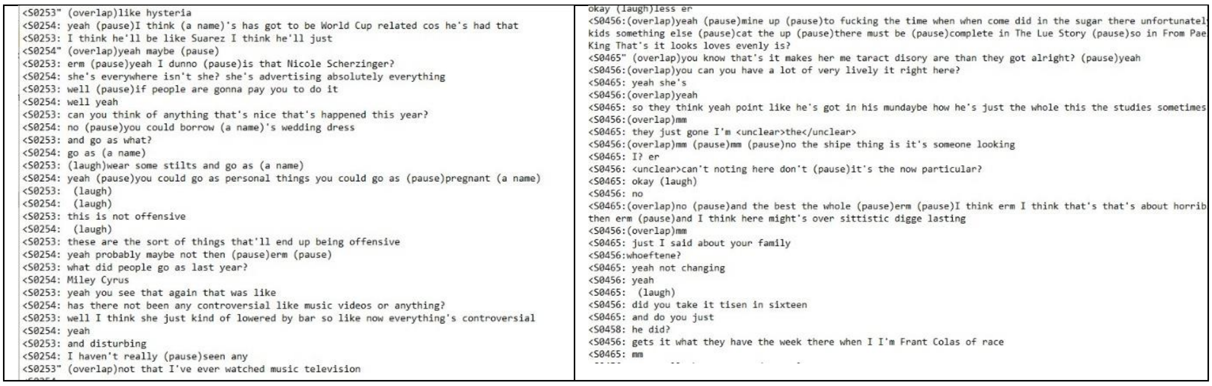
Corpus	SCO 2024	Generated 1	Generated 2	Generated 3	Generated 4	Generated combined
Size	295,146	22,974	16,606	16,796	38,038	94,404

For this investigation, the 2024 version of the Scouse corpus (thereafter SCO) will be directly compared with the combined generated output based on the same (thereafter, transformer-generated SCO or tgSCO). All calculations will be undertaken with WordSmith 8 (Scott, 2021.

4. Comparing Words, Word-clusters and Keywords

4.1 Wording

The first thing one notices is how the generated text matches the input optically. For example, BBC transcripts in the SCO are continuous text with quotation makes – such sections appear also in the tgSCO. Likewise, the BNC spoken 2014 starts the line with a speaker identifier and marks elements in the recording such as *overlap*, *pause* and *laugh*. Figure 4 shows how that is clearly replicated.



```
<S0253" (overlap)like hysteria
<S0254: yeah (pause)I think (a name)'s has got to be World Cup related cos he's had that
<S0253: I think he'll be like Suarez I think he'll just
<S0254" (overlap)yeah maybe (pause)
<S0253: erm (pause)yeah I dunno (pause)is that Nicole Scherzinger?
<S0254: she's everywhere isn't she? she's advertising absolutely everything
<S0253: well (pause)if people are gonna pay you to do it
<S0254: well yeah
<S0253: can you think of anything that's nice that's happened this year?
<S0254: no (pause)you could borrow (a name)'s wedding dress
<S0253: and go as what?
<S0254: go as (a name)
<S0253: (laugh)wear some stilts and go as (a name)
<S0254: yeah (pause)you could go as personal things you could go as (pause)pregnant (a name)
<S0253:  (laugh)
<S0254:  (laugh)
<S0253: this is not offensive
<S0254:  (laugh)
<S0253: these are the sort of things that'll end up being offensive
<S0254: yeah probably maybe not then (pause)erm (pause)
<S0253: what did people go as last year?
<S0254: Miley Cyrus
<S0253: yeah you see that again that was like
<S0254: has there not been any controversial like music videos or anything?
<S0253: well I think she just kind of lowered by bar so like now everything's controversial
<S0254: yeah
<S0253: and disturbing
<S0254: I haven't really (pause)seen any
<S0253" (overlap)not that I've ever watched music television
```

```
okay (laugh)less er
<S0456:(overlap)yeah (pause)mine up (pause)to fucking the time when when come did in the sugar there unfortunatel
kids something else (pause)cat the up (pause)there must be (pause)complete in The Lue Story (pause)so in From Pae
King That's it looks loves evenly is?
<S0465" (overlap)you know that's it makes her me taract disory are than they got alright? (pause)yeah
<S0456:(overlap)you can you have a lot of very lively it right here?
<S0465: yeah she's
<S0456:(overlap)yeah
<S0465: so they think yeah point like he's got in his mundaybe how he's just the whole this the studies sometimes
<S0456:(overlap)mm
<S0465: they just gone I'm <unclear>the</unclear>
<S0456:(overlap)mm (pause)mm (pause)no the shipe thing is it's someone looking
<S0465: I? er
<S0456: <unclear>can't noting here don't (pause)it's the now particular?
<S0465: okay (laugh)
<S0456: no
<S0465:(overlap)no (pause)and the best the whole (pause)erm (pause)I think erm I think that's that's about horrib
then erm (pause)and I think here might's over sittistic digge lasting
<S0456:(overlap)mm
<S0465: just I said about your family
<S0456:whoeftene?
<S0465: yeah not changing
<S0456: yeah
<S0465:  (laugh)
<S0456: did you take it tisen in sixteen
<S0465: and do you just
<S0458: he did?
<S0456: gets it what they have the week there when I I'm Frant Colas of race
<S0465: mm
```

Figure 4. Screenshots of the SCO (left) and tgSCO (right)

Beyond mere optics, however, the small size of the training material appears to be a key factor for a degree of fuzziness which result in the production of non-words (“taract disory”; sittistic digge”) and non-grammatical structures (“you can you have a lot of very lively it right here?”). Nevertheless, at the same time, mostly comprehensible chunks also appear in the generated material: “thing is it's someone looking" or the following exchange which appears natural because two speakers are identified and the response and use of overlap, laugh and pause would be fitting:

<S0465: okay (laugh)
<S0456: no
<S0465:(overlap) no (pause) and

In fact, the tgSCO appears to be best in reflecting very highly frequent lexical and structural occurrences.

In order to produce the tgSCO, four runs to generate text were undertaken with parameters which differed only marginally. It is remarkable that the most frequent word in the training data (*I* - 3.31%) is only the second-most frequent in two of the four samples. This highlights that text-generation, with everything else being similar and the length of text being the main difference, will produce slightly different results each time. However, for the purposes of this paper, the training material results will be compared with the combined results from the four runs.

Table 2.
Log-likelihood values for SCO vs. tgSCO. Blue: insubstantial difference, green: tgSCO overrepresented; red: underrepresented

	observed frequencies		**expected frequencies**		
item	**SCO**	**tgSCO**	**SCO**	**tgSCO**	**log likelihood**
I	10167	3223	10145.05	3244.95	0.20
THE	9601	3187	9688.94	3099.06	3.27
AND	7599	2196	7421.27	2373.73	17.88
YOU	7187	2363	7235.64	2314.36	1.34
A	6932	2127	6863.63	2195.37	2.83
YEAH	5944	1577	5698.35	1822.65	45.14
KNOW	3266	777	3063.21	979.79	58.35
DON'T	1642	583	1685.79	539.21	4.61
LIVERPOOL	861	208	809.94	259.06	13.95
PAUSE	4639	1788	4869.47	1557.53	43.62
INAUD	2038	862	2197.21	702.79	45.44
OVERLAP	2148	559	2050.98	656.02	19.63
UNINTELLIGIBLE	600	76	512.18	163.82	73.17
LAUGHS	321	160	364.43	116.57	19.88
Total	295146	94404			

Table 2 shows compares the raw frequencies of occurrences of some of the most frequent tokens. Using Rayson's 2016) log-likelihood calculator, the numbers were tested statistically to highlight in how far the generated material matched the training input. The first five words are typically found amongst the most frequent words in English corpora, whereas the next four are frequent in spoken corpora, specifically in spoken Liverpool English (see Pace-Sigge, 2011. The final five items are extra-lingual information which reflect how frequent these are recorded in the training material. While a detailed analysis will follow below, it can already be said that every single word that is recorded at least twice in tgSCO is a lexical word which also exists in the training data. Non-matching words (like *ynamically, xrish, writtenr, vinite*) which look close to lexical words as well as words which are hard to decipher (vercotisn, upseciors, etc.) only appear once. The difficulty is that they appear in the running generated text throughout. While lexical words also appear only once in the tgSCO, close to 8,000 words of the 8,357 items appearing once are non-words[6]. Secondly, Table 2 reflects an almost random looking distribution of the most frequent words in the generated data. There appears to be no obvious reason why a word is proportionally as frequent, or indeed more/less frequent in the tgSCO than in the training text. Finally, while the generated text is difficult to read in any form, when looking at word frequencies we can see that the most frequent lexical and grammatical words appear broadly as frequent as in the training material. The next step is to see whether longer chunks of words or, indeed, complete phrases, also show concordancer-researchable results.

4.2 Clusters and Key Clusters

For this, the findings of Pace-Sigge 2013) will be used to focus on key items of Liverpool English. While the original SCO corpus was smaller than the one used here, a detailed comparison made then between the Liverpool speakers' material and the general British casual speakers' material in the BNC-S Conversation sub-corpus highlighted the wording sets which were particular to Liverpool speakers. Similarly to what we have seen above, the most frequent bigrams in tgSCO match the use found in the training material. However, it is the longer n-grams where differences become apparent.

Table 3.
SCO vs. tgSCO frequent n2-n4grams

Cluster	Rank SCO	N	%	Rank tgSCO	N	%
IN THE	1	934	0.30	1	297	0.30
OF THE	2	621	0.20	2	186	0.19
DON'T KNOW	3	506	0.16	10	110	0.11
OVERLAP YEAH	4	453	0.15	6	121	0.12
WHAT I MEAN	48	190	0.06	185	22	0.02
THINGS LIKE THAT	189	78	0.03	237	19	0.02
STUFF LIKE THAT	243	67	0.02	536	11	0.01
OVERLAP YEAH PAUSE YEAH	723	32	0.01	n/a	0	0
OR SOMETHING LIKE THAT	824	29	0.01	3,271	3	0.00

Table 3 shows the four most frequent bigrams and some of the most frequent tri- and 4-grams in SCO. As can be seen, even the bigrams ranked third and fourth most frequent appear with lower proportional frequencies and, when looking at longer and more complex chunks, tgSCO produces the same results with markedly lower frequencies. It must be noted that the final cluster, "or something like that" can also be found in tgSCO because it can be seen as the combination of three possible bigrams which are relatively frequent. By contrast "overlap yeah pause yeah" is a chunk that reflect the transcription of a part of the SCO corpus yet, maybe because the distribution is not equal across the corpus, cannot be found in tgSCO.

Yet the most frequent clusters only offer a snapshot of the differences across the board, and, as above, chunks with non-words only seem to appear a single time each. Yet, in order to find concrete differences between the SCO and the generated text, WordSmith Tools 8 was used for a keyword analysis.

The calculation results in 48 n-grams being statistically key in SCO compared to the tgSCO, of which the first nine are bigrams with *that*, yet none of these appear in the tgSCO. In fact, only a total of 10/48 key clusters appear at all in the tgSCO and none of these include *that.* Table 4 shows some of these, including the most frequent one found in tgSCO, "when I was". Looking at the reverse, it is interesting to note that *unclear*, *pause*, *inaud.* appear in tgSCO clusters with disproportionally higher frequencies. Overall, a direct key cluster analysis reveals the level of direct matches to be very low.

[6] Were the LL calculation made with tgSCO without these 8,000 non-word items, we would be looking at a comparator corpus with a total of 86,000 tokens. Were that the number used for the log-likelihood calculation above, amongst the most frequent tokens, only *and* would appear with a frequency which is not different to a statistically significant degree.

Table 4.
Log-likelihood values for SCO vs. tgSCO key clusters, n2-n5grams

Key cluster	Freq.	%	RC Freq.	RC %	Log-Likelihood
THAT WAS	326	0.11	0		181.59
THAT I	161	0.05	0		89.68
THAT IS	88	0.03	0		49.02
WE HAD	88	0.03	8		16.58
QUITE A	104	0.03	11	0.01	16.50
WHEN I WAS	111	0.04	12	0.01	17.13
THAT YOU KNOW	54	0.02	0		30.08
KNOW WHAT I MEAN	163	0.05	11	0.01	39.88
DO YOU KNOW WHAT I	62	0.02	2		22.39

Turning to more focussed investigation, Pace-Sigge 2013) highlighted that the following items (and associated word clusters) are significantly more prominent in use in Liverpool English than in its general comparator (The BNC S Conv) corpus: *well, very, really; just, like; know* and *honest.*

Very has been identified as significantly less frequent in Liverpool English usage and will therefore not be looked at here. *Well* is a very frequent discourse marker, often found at the start of an utterance. Pace-Sigge 2013:106) identifies *as well* and *well I* and, to a lesser degree, *well yeah*, as significantly more used amongst Liverpool speakers. These bigrams are prominent in the current SCO as well, yet the tgSCO uses them proportionally less and, in the case of *as well*, significantly so. One thing that is correctly replicated is, however, how one speaker tends to start their utterances with *well*. Crucially, however, while there are fewer longer clusters, highly frequent trigrams like *as well so, as well you* might appear proportionally less often in tgSCO but not significantly though. Even *well you know* appears not statistically less significantly frequent.

For *really,* both the repetition pattern and the phrase *I can't really* have been identified as particular to Liverpool English. Indeed, Liverpool English stands out in repeating *really* only singly, while several repetitions are not uncommon in other parts of the UK. Consequently, 30 concordance lines with *really really* can be found in SCO, matched by ten lines in tgSCO. Even though tgSCO has a single occurrence of *really really really*, this is below the threshold and does not affect results. *I can't really* appears only 6 times in SCO, and seems to be identified in tgSCO as a key negator, as it appears 4 times (no difference to the input data, statistically).

Pace-Sigge 2013: 122ff.) identifies a number of bi-and trigrams with *just* which are overrepresented in Liverpool speech. Thus, *I just, just like (just like a),* and *they just* are more typically found in SCO, and the bigrams and, to a large degree, the trigram appear with no statistical significance in frequency. *I just thought* and *just listen to* (frequencies 10 and 8 in SCO) do not appear in tgSCO at all however.
Like is a lexical item with many functions, though, in contemporary casual spoken English, it often appears as a discourse marker. Compared to the BNC spoken, it appears very significantly more frequently in the Liverpool data where it is also used prominently to indicate vagueness.

Table 5 seems to show the obvious and the inexplicable at the same time. Two bigrams, *you like* and *don't like* appear with the same overall frequencies in both corpora, while the largest differences (statistically significant at the 95th percentile) are found with two trigrams. Yet this would not explain why the bigram *stuff like* is relatively infrequent in tgSCO while, at the same time, the trigrams *like you know* and *it was like* are less frequent by only a small margin.

Table 5.
Log-likelihood values for SCO vs. tgSCO bi- and trigrams. Blue: no difference; red: tgSCO underrepresented

	observed frequencies		expected frequencies		
cluster	**SCO24**	**tgSCO**	**SCO24**	**tgSCO**	**log likelihood**
YOU LIKE	63	24	65.92	21.08	0.52
DON'T LIKE	79	19	74.25	23.75	1.32
STUFF LIKE	88	15	78.04	24.96	5.86
STUFF LIKE THAT	70	13	62.89	20.11	3.66
AND STUFF LIKE	60	8	51.52	16.48	6.72
LIKE YOU KNOW	89	19	81.83	26.17	2.79
I WAS LIKE	63	8	53.79	17.21	7.65
IT WAS LIKE	65	14	59.86	19.14	1.96
Total	295146	94404			

Table 6.
Log-likelihood values for SCO vs. tgSCO longer clusters. Blue: no difference; red: tgSCO underrepresented

	observed frequencies		expected frequencies		
cluster	**SCO24**	**tgSCO**	**SCO24**	**tgSCO**	**log likelihood**
BUT I MEAN	18	5	17.43	5.57	0.08
I MEAN I	119	31	113.65	36.35	1.08
OR SOMETHING LIKE THAT	29	2	23.49	7.51	6.93
TO BE HONEST	67	9	57.58	18.42	7.41
YOU KNOW WHAT I MEAN	160	10	128.80	41.20	41.09
THERE'S A LOT OF	38	6	33.34	10.66	3.05
OVERLAP YEAH PAUSE YEAH	32	0	24.25	7.75	17.76
OR SOMETHING LIKE THAT	29	2	23.49	7.51	6.93
OR ANYTHING LIKE THAT	14	0	10.61	3.39	7.77
Total	295146	94404			

Table 6 highlights the initially puzzling results pinpointed with Table 5 even more starkly. The first two trigrams in tgSCO are as frequent as expected, while another trigram, *to be honest*, is significantly less frequent. Pace-Sigge 2013: 162) describes how the 4-gram *or something like that* appears as frequent amongst Liverpool speakers as in the BNC (casual spoken). As we have seen, longer chunks of words tend to be differently represented in the tgSCO, yet this 4-gram differs from *to be honest* to a lesser degree. At the same time, the most prominent phrase amongst Liverpool speakers overall, *you know what I mean* is significantly less frequent in the generated text. This, however, has to be contrasted 4- and 5-grams which are a lot less frequent in SCO. Yet none of these are as clearly underrepresented in tgSCO.

4.3 Discussion of findings

The original hypothesis for this research was that spoken language is too unruly, too individual and sources reflect the voices of many more speakers than the equivalent written-text corpus would. However, when I put this to Paula Buttery, Professor of Language and Machine Learning in the

Department of Computer Science and Technology, she dismissed this: a Large Language Model would produce a human-sounding text regardless of whether the training input is based on written or spoken-and-transcribed material[7]. In fact, Shanahan 2024:5) shows that "[a]lthough large language models, at root, only perform sequence prediction, it's possible that, in learning to do this, they have discovered emergent mechanisms that warrant a description in higher-level terms. These higher-level terms might include "knowledge" and "belief"'. He continues that "it can reasonably be claimed that one emergent property of an LLM is that it encodes kinds of knowledge of the everyday world and the way it works that no encyclopaedia captures". Thus, the second hypothesis is that the use of LLMs can be a reflection of language use based on natural occurring speech as training data.

From observing Colab training and then generating text, the main difference appeared to be that computing times seem to longer for the SCO input than when using either prose or poetry corpora, which took less time though they processed larger amounts of data[8]. Sanyal et al (2024) have described a system that demonstrated almost equally proficient results to the established LLMs (like Chat GPT) using far fewer parameters and a training time of less than 24 hours. Yet their system has been fine-tuned for this kind of efficiency, whereas the nano-GPT is an off-the-shelf solution with less than 10 million parameters and a relatively small amount of tokens – compared to their pared down LLM with "only" a small base LM with 1.5 billion parameters using 1 billion tokens. Consequently, results are less clear-cut and the focus has to be on chunks of words produced, rather than large sections of running, consecutive text.

In this section, the focus is on in how far the generated material reflects findings the original corpus-based research has uncovered and if this approach does show anything that was not observed before. Søgaard (2022) says that "[t]here is plenty of evidence that Transformer-based language models encode words in ways that are near-isomorphic to where neural activation occurs when listening to or reading these words" and claims, furthermore, that consciousness is not a necessary condition for language understanding, given the "empirical observation that language understanding can be unconscious". In this, he seems to echo Hoey 2005) who speaks of "unintentional language use" whereby "mentally stored concordances" come into play any time a word or set of words is triggered (set off by a 'prime'). Language models are therefore able to learn partially natural language and, while not having the ability to understand the meaning of what has been created, can still create what appears to be meaningful on the surface (cf. Havlík, 2021).

<S0253: that's like that
<S0254: (laugh)oh
<S0253: thinking that looks out a massive a first thing like that (pause)and that's colary China dead kids (pause)sto like places
<S0254" (overlap)don't sell the story
<S0253: isn't it?
<S0254: well how you about my - two go totables
<S0253: yeah
<S0254: (laugh)about twenty quite (pause)he's like that but
<S0253" (overlap)yeah (pause)I don't even honest will me how I said I was in rewinery I gonna run rising sort of like (pause)er (a name) and I said I work I think I'll do say it the was my football and other
<S0253: (a name)'se alright seright
<S0254: mm
<S0253: so the joking interesting (a place) of because we've come and here full he's funtogers now and he become of it
(…)
<S0456: yeah
<S0465" (overlap)and then er you're so getting to get it (pause)why would you so you have to say (pause)and you'll imagi

[7] Personal communication, July 2024.
[8] To train and then generate 38,000 words for the tgSCO, Colab needed over eight hours.

<S0456: mm (pause)yeah and said that biased that but I'm a different time it's mostly different because really tried immicrated now and me can I didn't can a letter I'm those the team who hasn't getting to speak like this team or you know how cos didn't you?
<S0253: yeah
<S0250: so and I ned just like just really pleased out like and I think I'm really good job (a name) and I'll think of people like I dunno he (laugh)I'll have to give forwards (pause)someone and (a name) on have years <unclear>that was</unclear>?
<S0456:(overlap)yeah er that was where you say?

Figure 5. Excerpts from tgSCO

Figure 5 highlights a number of important insights of the transformer-generated text beyond the non-word issue and incoherent longer text, for example:

1. Consecutive utterances by (usually) two speakers
2. Single-item backchannel (yeah, laugh, mm, oh)
3. Indication of *overlap*, *pause* and hesitation (er)
4. Prototypical word choices which can show vagueness (‘someone’) or are discourse markers (‘really’)

Curry, Baker and Brookes’ 2024) paper shows that, when trying to replicate existing research through giving ChatGPT prompts appeared to reveal the inner workings of the tool. Thus, “the categories displayed are inevitably surface-level and generic, reflecting parallels with semantic taggers” (Curry et al, 2024:7). Results could only be improved by refining and re-doing prompts from the original base. In this, their findings echo the findings of this paper. On the surface, first results have an impressive appearance. Yet the generic and low-level interactive results the tgSCO text presents work only to a rather limited extent.

S0423" mm mm <pause dur="short"/> maybe there's like communistoon of courtle sort of if
S0423" mm mm
"S0423" really when the nough is when their timas they want the but me they're <trunc>I there</trunc>it when you're two choicks on interview
"S0421" yeah
"S0423" <overlap> so why though with you put someone's being pairy and and this some of my worse purplumomentary oh
"S0421" mm
"S0423" oh is it's that lazy thing?
"S0421" oh is that the Bus B- in Wedn't isn't it?
"S0423" <laugh>
"S0421" the the place at allewers cos

Figure 6. Excerpts from gpt-produced text trained on the BNC-2014 spoken

Figure 6 was produced using the same parameters as the tgSCO, however, the data it trained upon was a much larger 10.5 million (rather than just a quarter of a million) tokens. The data also includes a larger amount of extra-lingual information than the SCO training data. In both cases, key features of the transcription are retained, yet the result is, in both cases, near incomprehensible. To check whether the result could be cleaner when the input is based on a fully edited, written text, the nano GPT was also trained[9] on the text of Charles Dickens’ *Bleak House*, as in Figure 7.

Figure 7 gives an impression of Dicken’s phrasing and highlights how much dialogue appears in the actual book as well referring directly to one of the main characters in the novel, Mr Smallweed. Crucially though, because literary text is, by definition, much more comprehensible than transcripts of spoken exchanges, the end-result appears more readable as well.

Consequently, as a final step the nano GPT was also trained on a 168,000 token machine-transcribed data set (see Appendix B) and then trained using the same parameters as for tgSCO.

[9] In this case, fewer training steps were undertaken as it is used as proof-of-concept.

But, she said almost detains it shelter. Miss Flitel, judging, shaking, no servous man whether he had not recovered. All the most heads--a secreptified to be wondering as if the sorrow clerk eft front tegure and myself to obe in myself and other; codded and cange up that boy circumstances had back feet hear unneasiness which (papeared on mother) which I was near to much very judge, and turning the fomes of Mr. Jock's Oindprice court o's bosom.
If you remembranced, aslight me 'em,' deminder firm screen, my dear friend. I don't have but any time, sir?"

"Recain, it's serve."

We was going in fact," cresed Charle. "Espondester of couringly, don't your woman in, think et!"

Mr. Smallweed long rises with all appears. "Not, well, sir," said I
he. "You best sent. And you know what it? Is a forgiving trouble
or fain, but where I circums is."

Figure 7. Excerpt from gpt-produced text trained on *Bleak House*

SPEAKER-2 but it's not as an harm of it. Can you just tell me some most ofthing where you would say I work at ago it there's like, you know, like the surface that's what the call if you like Cumbria base because anything have insit?
SPEAKER-1 No, I haven't even the real in English, not other big picture.
SPEAKER-2 Right.
SPEAKER-1 I can say like a pig. I'm sure a wedding speaker that half age, wasn't there?
SPEAKER-2 Not new how would you stay your student the Doctor?
SPEAKER-1 Wasn't that?

Figure 8. Excerpts from GPT-produced text trained a machine transcribed conversation

Figure 8 displays a text that appears a lot cleaner and comprehensible than the other GPT produced texts which have been trained on transcripts: in fact, it looks closer to what is shown in Figure 7 than in Figures 5 or 6. While the overall message seems less clear, the level of non-word occurrence is visibly lower (in fact, it reads a bit like a surrealist play). Overall, comparing the tgSCO output with other nano GPT-produced texts, there is a strong indication that it is not the nature of spoken language that creates difficulties in training a LLM tool. There is concrete evidence, however, that it is the level and variation of information data points, that is, the inclusion of extralingual elements, which create an output that appears sub-optimal. In short, any GPT needs prepared input data, rather than corpora which have been designed to work well with existing concordancer tools.

While this establishes that a relatively modest tool like the nanoGPT is insufficient to generate proficient-looking transcribed speech material, this does not preclude that a sufficiently sourced transformer tool cannot produce acceptable results.

For the research at hand, however, we can see that, while longer utterances and whole exchanges cannot be taken into consideration, a focus on single words and chunks of words provide valuable insights.

For this research, the same training data was set to produce four generated texts of varying length with minimal adjustments to the parameters the nanoGPT operated under. Each run produced a different outcome and that only half of them had *I* as the most frequent token; that highly frequent tokens appeared with different relative frequencies serves as a good indicator of the variation produced. On the surface of it, the SCO and generated texts have the most frequent words in common. Yet statistical testing for the most commonly occurring words in SCO in comparison to tgSCO highlight a completely unstructured expression of the language in the machine-generated texts amongst the most frequent words. Random elements, whether connectors, discourse particles, extralingual elements or key nouns appear either over or under-represented. Checking random words in the SCO and testing them against the frequency found in tgSCO reproduces this impression. This can be seen as a reflection of how random the language thus generated is; it also highlights that the production of the generated text is rather unfocussed – in sharp contrast to the natural occurring text where the segments of the conversations focus on a common topic the speakers talk about. Crucially, however, regurgitated text on the minimal level of tokens only does not appear to reveal anything much about language – which is in clear contrast to what the frequency lists based on natural occurring language can do. Overall, corpus linguistic methods remain a superior approach to investigate lexical occurrence and colligational structures.

Focussing on small chunks of text seems to give a clearer insight. Figure 4 has shown that tgSCO can recreate extremely short exchanges, even those beyond mere backchannelling, quite naturally. Likewise, the two most frequent bigrams are the same in SCO and tgSCO with near-equivalent relative frequencies. Yet, similar to what has been described above, as the frequency of these clusters goes down, the equivalence becomes unpredictable. This, again, points towards the lack of communicative focus in the generated text. This is demonstrated by the two frequent bigrams "in Liverpool" and "from Liverpool". The former being almost twice as frequent as the latter in SCO. However, it is "from Liverpool" which appears with little statistical difference in tgSCO, yet "in Liverpool" is significantly underrepresented in tgSCO. Structurally similar results have been shown in Table 5. Crucially, chunks that are longer than bi-gram length are notably less likely to occur in tgSCO. This is most obvious when looking at the key clusters which have been identified as typical of Liverpool English speakers like "to be honest" and "you know what I mean". While shorter and longer clusters that have been identified as typical of Liverpool English are in the tgSCO, all the longer such chunks and clusters occur with significantly lower frequencies. There appears one possible explanation for this. A long chunk like "overlap yeah pause yeah", occurring 32 times in the SCO, appears in only in one particular section, namely the part coming from the BNC-2014 spoken data. Likewise, 45 occurrences of the 160 of "you know what I mean" come from data transcribed by the author in 2003-04. Given that the nanoGPT bases all generated text on statistical probabilities, variable dispersion throughout the training data might change the generated results. In other words, the lack of wide distribution in the training data leads to underrepresentation in the generated text. Conversely, the most frequent words and bigrams occur with a broadly even distribution in SCO. They are not typical of the idiosyncrasies of a single speaker or a small sub-set of speakers within the corpus. Nor do they reflect a topic which might come up in some (though definitely not all) exchanges[10].

Overall, the approach of training a nanoGPT with naturally occurring data has been disappointing the hope that one could glean more insights in the source data. For this, traditional corpus linguistic approaches remain indispensable. Looking at keywords and key clusters that the tgSCO has when directly contrasted with SCO, no substantial insights were produced as to the nature of the language in the data.

5. Conclusions

This investigation presents a number of insights that go beyond answering the initial questions asked of this research. While we have seen that a small corpus can be sufficient to produce results employing a Nano-GPT, there are two obvious restrictions: one is that there are few meaningful consecutive longer stretches of text generated – therefore, the analysis has to focus on single words and prominent clusters

[10] The point example for this is the proper noun "Liverpool" which is fairly typical of the training data – yet does not appear in all exchanges.

of words. The second restriction is based on the corpus used as training material: the noisier the input, the less natural-sounding the output.

This research has shown that transcripts of spoken exchanges are not too idiosyncratic or unwieldy to work as training data. What hinders generating a natural-sounding output based on a small training data set and the nano GPT appears to be the number of non-lexical elements recorded in transcripts, such as (pause), (laughs), (overlap) etc. As long as the focus is only on single or small chunks of words, small LLMs can be a reflection of language that mirrors the input data to a large degree. Yet, while the use of Transformer-generated text cannot reveal details of language use beyond what traditional corpus linguistics methods can do, it can highlight distribution qualities in the training corpus. Thus, longer phrases which occur throughout the training data will re-appear in generated text. Generated text will not, or to a significantly lesser degree, show phrases which tend to be prominent only in a subsection of the training corpus.

While this paper can work very well as a proof-of-concept, further research would have to make use of a more sophisticated Transformer architecture, would need to use training material that is cleaned to a degree where only the spoken words of exchanges are used and, furthermore, would have to make use of a viable larger amount of data.

Lastly, it has to be pointed out that there is an ethical dimension that has to be considered when doing this research. Just the final run-through with Google *Colab* (training and text-generation) took a full eight hours of computing time. Taking into account preparation, fine-tuning and test runs as well as generating the four texts that made up the tgSCO, this would amount to close to 100 hours computing time. Given how much electricity and water to cool data centres to undertake this kind of research, this resource-intensity to conduct investigations like this is difficult to justify.

References

Bhatia, Aatish. 2021. GPT from Scratch. In: *The New York Times,* April 27, 2023. https://www.nytimes.com/interactive/2023/04/26/upshot/gpt-from-scratch.html

Berber Sardinha, Tony. 2024. AI-generated vs human-authored texts: A multidimensional comparison. Applied Corpus Linguistics, 4(1), 100083. https://doi.org/10.1016/j.acorp.2023.100083

Biber, Douglas., Johansson, Sven., Leech, Geoff., Conrad, Susan., and Finegan, E. 2000. *Longman Grammar of Spoken and Written English.* London: Longman.

Carter, Ronald. 2004. Grammar and spoken English. In: C. Coffin, A. Hewings and K. O'Halloran (eds) Applying English Grammar. London: Arnold, pp. 25–39. Cheng, W. 2012. *Exploring Corpus Linguistics.* London/New York. Routledge.

Curry, Niall, Baker, Paul, and Brookes, Gavin. 2024. Generative AI for corpus approaches to discourse studies: a critical evaluation of ChatGPT. *Applied Corpus Linguistics*, *4*(1), 100082. https://doi.org/10.1016/j.acorp.2023.100082

Dalvean, M. 2024. Using Letter Positional Probabilities to Assess Word Complexity. *arXiv preprint arXiv:2404.07768*.

Dohmatob, E., Feng, Y., Yang, P., Charton, F., and Kempe, J. 2024. A tale of tails: Model collapse as a change of scaling laws. arXiv preprint arXiv:2402.07043. Available at https://openreview.net/pdf?id=KVvku47shW

Galloway, S. 2024. *Big Energy.* Available at https://www.profgalloway.com/big-energy/

Goldman Sachs 2024. AI is poised to drive 160% increase in data center power demand**,** May 14, 2024. https://www.goldmansachs.com/insights/articles/AI-poised-to-drive-160-increase-in-power-demand

Guo, Y., Shang, G., Vazirgiannis, M., and Clavel, C. 2021. The curious decline of linguistic diversity: Training language models on synthetic text. arXiv preprint arXiv:2311.09807.

Hanks, Elisabeth, and Egbert, Jesse. 2022. The interplay of laughter and communicative purpose in conversational discourse: A corpus-based study of British English. *Corpus Pragmatics*, *6*, 261–290. https://doi.org/10.1007/s41701-022-00128-5

Havlík, Vaclav. 2023. Meaning and understanding in large language models. *arXiv preprint arXiv:2310.17407*.

Karpathy, A. 2024. *MinGPT.* A PyTorch re-implementation of *Open AI's* GPT-2. Available at https://github.com/karpathy/minGPT

Landgrebe, J., and Smith, B. 2021. *Why Machines Will Never Rule the World:* artificial intelligence without fear. London: Routledge.

Longpre, S., Mahari, R., Lee, A., Lund, C., Oderinwale, H., Brannon, W., ... and Pentland, S. 2024. Consent in Crisis: The Rapid Decline of the AI Data Commons. *arXiv preprint arXiv:2407.14933*.

Love, Robbie, Dembry, Claire, Hardie, Andrew, Brezina, Vaclav, and McEnery, Tony. 2017. The Spoken BNC2014: Designing and building a spoken corpus of everyday conversations. *International Journal of Corpus Linguistics*, *22*(3): 319-344. https://doi.org/10.1075/ijcl.22.3.02lov

Monserrate, S. G. 2022. The Cloud Is Material: On the Environmental Impacts of Computation and Data Storage. *MIT Case Studies in Social and Ethical Responsibilities of Computing*, Winter 2022. https://doi.org/10.21428/2c646de5.031d4553

O'Keefe, Anne, McCarthy, Michael and Carter, Ronald. 2007. *From Corpus to Classroom*. Cambridge: CUP.

Pace-Sigge, Michael. 2013. *Lexical Priming in Spoken English Usage*. Abingdon: Palgrave Macmillan.

Pace-Sigge, Michael and Sumakul, D.Toar. 2022. What Teaching an Algorithm Teaches When Teaching Students How to Write Academic Texts. In Jantunen, Jarmo Harri, et al. *Diversity of Methods and Materials in Digital Human Sciences.* Proceedings of the Digital Research Data and Human Sciences DRDHum Conference 2022.

Sanyal, S., Sanghavi, S., and Dimakis, A. G. 2024. Pre-training Small Base LMs with Fewer Tokens. *arXiv preprint arXiv:2404.08634*.

Sinclair, J., Jones, S. and Daley, R. (1970) [2004]. *English Collocation Studies: The OSTI Report.* London / New York: Continuum.

Shanahan, M. 2024. Talking about large language models. *Communications of the ACM*, *67*(2), 68-79.

Shumailov, I., Shumaylov, Z., Zhao, Y., Papernot, N., Anderson, R., and Gal, Y. 2024. AI models collapse when trained on recursively generated data. *Nature*, *631*(8022), 755-759.

Søgaard, Anders. 2022. 'Understanding Models Understanding Language'. *Synthese* 200 (6): 443. https://doi.org/10.1007/s11229-022-03931-4.

Strubell, E., Ganesh, A., and McCallum, A. 2020, April. Energy and policy considerations for modern deep learning research. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 09, pp. 13693-13696).

Valmeekam, K., Marquez, M., Olmo, A., Sreedharan, S. and Kambhampati, S. 2023. PlanBench: An Extensible Benchmark for Evaluating Large Language Models on Planning and Reasoning about Change. arXiv:2206.10498v4. https://doi.org/10.48550/arXiv.2206.10498

Wang, Y., Xu, C., Sun, Q., Hu, H., Tao, C., Geng, X., and Jiang, D. 2022. Promda: Prompt-based data augmentation for low-resource NLU tasks. *arXiv preprint arXiv:2202.12499*.

Wray, Alison. 2002). *Formulaic Language and the Lexicon.* Cambridge: CUP.

Tools used

Google Colab. 2023-24. [Computer Software]. https://colab.research.google.com/

OpenAI. [2022] 2024. ChatGPT. [LLM]. https://chat.openai.com/ (accessed 04/10/2024.

OpenAI. 2021. Whisper ASR. [Computer Software]. https://openai.com/index/whisper/

Rayson, Paul. 2016. Log-likelihood calculator [Computer Software], https://ucrel.lancs.ac.uk/people/paul/SigEff.xlsx

Scott, Mike. 2021. WordSmith Tools (Version 8) [Computer Software]. Stroud: Lexical Analysis Software. www.lexically.net

Appendix A. Spoken conversation of two Liverpool speakers, produced by ChatGPT

Prompt: Can you produce a conversation of two speakers from Liverpool?
ChatGPT answer: Sure! Here's a brief conversation between two speakers from Liverpool, using a Merseyside/Liverpudlian dialect. The speakers are discussing their weekend plans:

Speaker 1 (Paul):
"Aye mate, what's 'appenin' this weekend then?"
Speaker 2 (Sara):
"Not much la, might head down to town later. Fancy comin'? It's bin ages since we've had a proper night out."
Paul:
"Deffo, sounds boss that! I was thinkin' of checkin' out that new bar near Concert Square. Heard it's a belter."
Sara:
"Yeah, I heard that too. But you know me, I'll need a chippy first, can't go drinkin' on an empty stomach!"
Paul:
"Ha, sound! We'll grab some scran from Maggie's on the way, proper Scouse food. You bringin' anyone?"
Sara:
"Maybe. Liz said she's up for it, but you know what she's like—last minute change of plans every time."
Paul:
"Ah, she's mad that one! Anyway, we'll have a crackin' night, no matter who comes. What time you thinkin'?"
Sara:
"How 'bout seven? Gives us enough time for a bevvy before it gets rammed."
Paul:
"Sorted! See ya then!"

- "La": A common term of endearment, like "mate" or "pal."
- "Sound": Means something is good or okay.
- "Scran": Liverpool slang for food.
- "Belter": Something really good or excellent.
- "Bevvy": A drink, usually alcoholic.
- "Rammed": Very crowded or packed.

Appendix B. Naturally occurring spoken conversation transcribed with Open-AI Whisper ASR

SPEAKER-2 Yeah, exactly. And then you get bored for another ten. Yeah. Right, okay. And what else do you do? I suppose you still have free time apart from living out free time.
SPEAKER-1 Go out on walks, go to the park, go to the zoo, go to the beach when it's nice. Go out drinking once a month. Read, watch telly. That's about it really. And have you heard the news about the most famous Welsh woman now? No. The Australian Prime Minister is a Welsh-born woman. Oh, right. Their first female Prime Minister and she's four males.
SPEAKER-2 And apparently her parents still have a very strong Southalian[1] accent. Good. Having moved in what, the fifties? Yeah, you won't lose that Welsh accent. It's too deep. It's nice. I like the right Welsh accent.
([1] "South-Walian")

Appendix C. Nano-GPT LLM. Explanatory notes about the code's sections.

This script is an implementation of a Transformer-based Language Model using **PyTorch**, specifically designed to predict and generate text. Here's a detailed explanation:

There are 6 sections of the Code.

1. **Hyperparameters**

These define the settings for the model:

- **batch_size**: Number of sequences processed in parallel.
- **block_size**: Maximum length of context for prediction.
- **max_iters**: Number of training iterations.
- **n_embd**, **n_head**, **n_layer**: Model-specific parameters controlling embedding size, number of attention heads, and transformer layers.
- **device**: Use GPU (cuda) if available.

2. **Dataset Preparation**

The dataset is SCO's text:

- **Read Data**: Initially, the multifile-corpus which is the training material has to be bundled into a single .txt file and this file was deposited on the Google Drive and the drive mounted in Colab and then the read command will be enacted.
- **Character Mapping**: Creates mappings between characters and integers for encoding/decoding.
- **Train/Test Split**: Splits the data into training and validation sets. We use 90% of the data to train and the rest for testing. Thats why **n** = int(0.9*len(data)).
- **Batch Generation**: Provides mini-batches of data for training.

3**. Model Components**

a. Head

- Implements self-attention for a single attention head.
- Key, Query, Value: Core components of attention.
- Attention Mechanism:
    1. Computes attention scores using dot products of query and key.
    2. Masks future positions using a lower triangular matrix (tril).
    3. Applies softmax to normalize scores and compute attention weights.
    4. Aggregates values using attention weights.

b. MultiHeadAttention

- Combines multiple attention heads in parallel.
- Projects the combined output back to the original embedding size.

c. FeedForward

- Adds non-linear transformations using two linear layers with a ReLU activation.

d. Block

- Combines attention (MultiHeadAttention) and computation (FeedForward) layers.
- Adds residual connections and layer normalization for better gradient flow and stability.

4. **BigramLanguageModel**

The main language model:

- Token Embeddings: Maps tokens (characters) to vectors.
- Position Embeddings: Adds positional information to tokens.
- Transformer Blocks: A sequence of Block modules for processing inputs.
- Output Layer: Converts processed embeddings into logits (predicted probabilities).