IT UNIVERSITY OF COPENHAGEN

Leveraging LLMs to create semantically enriched citation networks

Mads Høgenhaug, Marcus Friis, Morten Pedersen

Research project , M.Sc. 7.5 ECTS
Course code: KIREPRO1PE
Data Science
Mads Høgenhaug, Marcus Friis, Morten Pedersen

December 15, 2023
Supervisor: Luca Rossi

ABSTRACT

With large language models such as ChatGPT, Llama2 etc., new opportunities arise. One such opportunity is leveraging the technology to aid understanding of existing problems. In this study, we use LLMs to augment an existing dataset to facilitate a deeper understanding. Specifically, we augment the Arxiv Hep-Ph citation network by: **a)** adding node attributes in the form of abstract embeddings using Llama2 13B, and **b)** labelling every edge in the network with a semantic label of agreement using `gpt-3.5-turbo`. With this approach, we create a new augmented dataset: *Cit-Hep-Ph-Aug*. Analyzing this, we find that, while the core augmentation method has potential, this specific implementation is lacking, and the target label is ambiguous, leading to a subpar dataset. To improve the approach, one could use better models, and redefine the target labels.

All data and code is available at
https://github.com/Marcus-Friis/research-project/

CONTENTS

Abstract i
Preface iii
1 Introduction 1
2 Background 2
3 Data and Material 3
3.1 Basic network properties . . . 3
3.2 Arxiv data collection . . . 4
4 Method 5
4.1 Augmenting the Hep-Ph citation network . . . 5
4.1.1 Abstract Embeddings . . . 5
4.1.2 Edge Classification . . . 5
4.1.2.1 Model Selection . . . 6
4.1.2.2 Prompt Engineering . . . 6
4.1.2.3 Post Collection Cleaning . . . 6
4.2 Edge Label Quality Assessment . . . 7
4.2.1 Quantitative . . . 7
4.2.2 Qualitative . . . 7
4.3 Analysis of data . . . 7
4.3.1 Community Detection . . . 8
4.3.1.1 Multilayer Community Detection . . . 8
4.3.2 Clustering . . . 9
4.3.3 Partition Comparison . . . 9
4.3.4 Homophily with Edge Labels and Node Attributes . . . 9
5 Results 10
5.1 Dataset presentation . . . 10
5.2 Edge label quality . . . 11
5.3 Analysis of dataset . . . 14
6 Analysis 17
6.1 Problems of Agreement/Neutral/disagreement . . . 17
6.1.1 Qualitative Analysis . . . 17
6.1.2 Quantitative Analysis . . . 17
6.1.3 Ideal Balance of Labels . . . 18
6.2 Improving LLMs . . . 18
7 Conclusion 19
A Appendix 24
A.1 Self-loops . . . 24
A.2 Prompts . . . 24
A.3 Partition Metrics . . . 25

PREFACE

We would like to thank Luca Rossi for being a great supervisor, guiding us through the project. We would also like to express our gratitude to Michele Coscia for providing guidance in setting up the Llama2 locally.

The authors acknowledge the IT University of Copenhagen HPC resources made available for conducting the research reported in this paper.

1 INTRODUCTION

The introduction of large scale language models signifies a paradigm shift with a tremendous impact on society, business and research [1, 2, 3], despite its major shortcomings [4, 5]. In research applications, it can be used to enhance understanding and augment existing research [6]. One such application is with citation networks. Citation networks have been extensively studied in traditional network analysis settings [7, 8, 9, 10]. However, using state-of-the-art large language models, we leverage their understanding of natural language to augment an existing dataset. Throughout this paper the term "LLM" or "LLM chat bots" will be frequently used, referring only to large language models such as GPT, BARD, or Llama.

In this paper, we aim to deepen our understanding of citation networks as well as provide an example of how LLMs in practice can be used as a tool for expanding research capabilities. To this end, we use the Arxiv Hep-Ph citation network [11, 7], which is a directed network where each node is a high energy physics - phenomenology paper, and each edge is a citation from one paper to another. We collect the abstract for each node, use the abstract and LLMs for: **a)** getting node embeddings, and **b)** getting edge labels based on the semantic relationship of a citation. This constitutes our main contribution. We present an augmented Hep-Ph citation network: *Cit-Hep-Ph-Aug*. With this augmented network, we demonstrate an example of how additional data facilitates new analyses, leveraging the newest movements in NLP.

The core of our analysis investigates homophily in the network. Homophily is a well known concept in networks used to describe how nodes that share similar attributes are more likely to connect to each other [12]. We explore how the topology of the network relates to the semantic meaning of each article, and the inter-article agreement. Are local structures at a mesoscale based on semantic homophily? Are there homophilic tendencies based on clusters of agreement? In practice, we analyse this by comparing traditional community detection with unsupervised clustering of article embeddings and multilayer community detection based on the semantic relation of a citation. On a micro scale, we aim to study how these different partitions deepen our understanding of citations, and how papers are grouped together. On a macro level, we study the validity of our approach; how LLMs can be used to enhance existing problems using embeddings and prompting.

While our analysis of Cit-Hep-Ph-Aug does not find any major discoveries, we demonstrate both a way of analysing such networks; but more importantly, we exemplify a scheme for enhancing networks with natural language attributes.

The rest of this paper elaborates on all of the details of previous mentioned procedures, and their findings. In 2 Background, we detail the most relevant similar research and the core methods we apply. Thereafter, in 3 Data and Material, we describe the Hep-Ph domain, present the most essential findings of a traditional network analysis, and detail the Arxiv data collection process. In the subsequent section (4 Method), we elaborate on the abstract embedding method and its unsupervised clustering. This includes edge labeling with ChatGPT, the application of multilayer community detection to the new edge layers, and a comprehensive comparison of various partitioning methods. The result of the described methods are presented in 5 Results, and then analysed and discussed in 6 Analysis, ending with a summary of the main findings and reflections in 7 Conclusion.

2 BACKGROUND

The Hep-Ph citation network, along with the similar Hep-Th citation network, originates from the 2003 KDD Cup [11]. In Leskovec et al. [7] they do a temporal analysis of how the network changes over time, showing that the number of edges grows superlinearly in the number of nodes, and the average distance between nodes shrinks over time. The data is widely used in other network science papers (e.g. [13, 14, 15]), typically in the context of applying methods on a real world network.

In our work, we aim to augment the citation network with labels of citation type. This task is similar to Kumar's *"Structure and dynamics of signed citation networks"* [10]. In this work, Kumar demonstrates that research paper's citations are not uniform and they can be categorized into three categories: Endorsement (positive), criticism (negative), and neutral. His work shows that positive citations occur twice as frequently as negative ones, postulating that papers tend to explicitly praise prior research that they build on. However, he mentions that a majority amount of citations fall under the 'neutral' category.

Kumar's methodology is simple; classifying the various citations into three categories based on a keyword-based technique derived from Wilson et al. [16]. He concludes the paper by mentioning that in order to get a better understanding of the sentiment, the study can be improved by applying more advanced NLP methods, instead of using keyword-based techniques. This is where LLMs come in handy, as some might say they are a bit more advanced than keyword-based techniques.

Another example of a similar task is Scite.ai. Scite.ai is a website that "display the context of the citation and describe whether the article provides supporting or contrasting evidence". As described in the paper about the citation index, it uses a deep learning model to categorize citations based on context, classifying as either "supporting", "contrasting" or "mentioning" [17]. This is similar our goal, and a practical application of using models to label edges, in order to get a deeper understanding of a citation network. Building upon Kumar's findings, and inspired by Scite.ai, our project aims to leverage LLMs to get a comprehensive understanding of citations by augmenting the Hep-Ph network with a semantic relation label.

Before we can leverage LLMs to label citations, its essential to get an understanding of how LLMs can process and understand human text. For this, we look to the paper by Zhang et al. [18] who conducted a comprehensive analysis into LLMs' capabilities for sentiment analysis. The study concludes that there is promising potential for LLMs, especially for simpler tasks and few-shot learning. Yet, when transitioning to more complex sentiment nuances, sentiment analysis poses a bigger challenge for the LLM. This is important to keep in mind, as the subject 'high energy physics' contains a lot of domain specific jargon; jargon an LLM might not have had much training data for. When using an LLM it is important to take into account that its answer to a question is largely based on how well defined the question is for a particular topic [19]. There are several methods that have been tested out by other researchers [20, 21, 22, 23] that gains marginal improvements across multiple tests. Instructing the model to act as a 'high energy physics expert' is a simple, quick, and cheap[1] way to alleviate some of the potential problems that could arise when working with papers of this caliber.

[1] Shorter custom instructions leads to fewer tokens used for each call, resulting in cheaper API calls.

3 DATA AND MATERIAL

The Hep-Ph citation network [11] is a citation network of High-energy physics papers (also known as particle physics) within the phenomenology category (a subfield which bridges the gap between mathematical models and experimental results). The network was originally released as a part of the 2003 KDD Cup, a knowledge discovery and data mining competition. It was mined from arXiv, an open-access repository of electronic preprints and postprints, and was created by "automated heurestics followed by human post-processing". The dataset spans from 1993 to 2003[1], effectively encompassing nearly the entire history of Hep-Ph up until 2003. It is not perfect due to spelling variations, abbreviations, typos etc., but it achieves "reasonable accuracy" [11]. The resulting dataset contains 34,546 papers (nodes) with 421,578 citations (edges). Some papers have multiple subcategories, but all have phenomenology as the primary category. The edges are directed from a paper to the paper it cites. Edges going outside the network are not included, whether its a paper inside our network that cites to one outside, or the opposite.

3.1 BASIC NETWORK PROPERTIES

Table 1: Overview of basic network metrics. All metrics marked with * are computed as undirected.

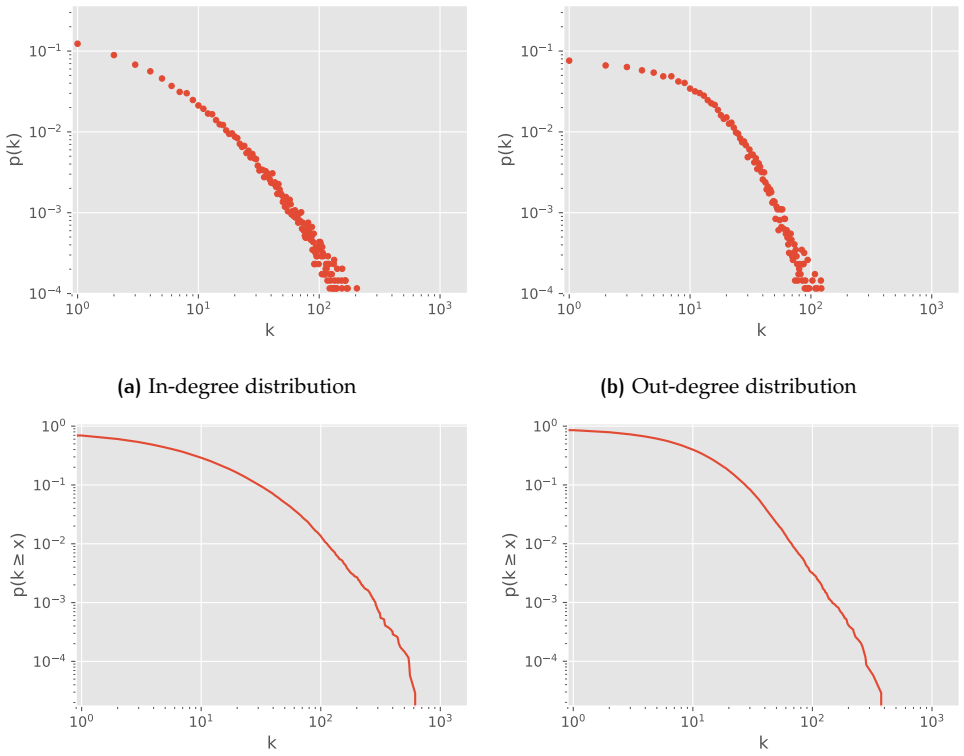
Metric	Value
Number of Nodes	34,546
Number of Edges	421,578
Average degree	24.4
Number of Self-loops	44
Average path length*	4.33
Diameter*	14
Number of Weakly Connected Components	61
Number of Strongly Connected Components	21,608
Average Clustering Coefficient*	0.14
Density of Network*	0.0004

As seen in Table 1, the network has $n = 34,546$ nodes and $m = 421,578$ edges. This results in a density $d = 0.0004$ meaning the network is sparse. When calculating the density, we treat the network as undirected, since, due to the nature of the domain, if $u \neq v$ and $(u, v) \in E$, then $(v, u) \notin E$. I.e., citations from node u to v imply there is no edge from v to u. Despite this property, the network is not acyclic, even without self-loops.

The network has 44 nodes with self-loops, each one with exactly one loop. By manual inspection, we see that some of them are self citations, while others seem to be an error. An exhaustive list of the nodes with self-loops can found in Appendix A.1.

The in- and out-degree distribution can be seen in figure 1. From this, it is evident that there is a longer tail for in-degrees than out-degrees. The max in-degree is $\max k_{in} = 846$, with the median in-degree being $\mathtt{median}\ k_{in} = 4$. For the out-degree, $\max k_{out} = 411$ and $\mathtt{median}\ k_{out} = 8$.

[1] Stanford claims the papers covers up to April 2003. However, upon closer look at the data, we find the latest papers are from 2002



(a) In-degree distribution

(b) Out-degree distribution

(c) Complementary cumulative in-degree distribution

(d) Complementary cumulative out-degree distribution

Figure 1: Various degree distribution plots. The top plots show the probability of each degree in a log-log plot. The bottom plots show the complementary cumulative distributions i.e. $p(k \geqslant x)$ the probability of a node having degree x or higher.

The network contains $21,608$ strongly connected components. However, strongly connected components are a peculiar concept in this network, given that citations cannot go forward in time. Ignoring direction, the network contains 61 weakly connected components, with the largest component containing $n = 34401, m = 421441$ nodes and edges, and the second largest containing $n = 6, m = 8$. For the analysis, we work strictly with the largest weakly connected component.

3.2 ARXIV DATA COLLECTION

A central part of this project is working with the abstract of nodes. These are not immediately available through the Hep-Ph network data. Instead, as part of the data augmentation process, they are collected through Arxiv's API. Through its API, the service provides data most notably regarding papers' abstract, authors, title, publish date, but also much more. The most essential part for this project is the abstract, but all the data the API offers has been collected for every paper within the dataset, and made publicly available on GitHub. The only exception is for two papers (with id `9812218` and `9305237`), which have been removed from the Hep-Ph section. Since abstracts are needed for the analysis, and they do not belong to Hep-Ph, these nodes are henceforth ignored for analysis purposes.

4 METHOD

This project includes several different methods for primarily three purposes; gathering/augmenting the existing Hep-Ph network, evaluating the quality of the edge labels, and an analysis of the resulting dataset focused on various unsupervised partitioning methods.

4.1 AUGMENTING THE HEP-PH CITATION NETWORK

On top of the collection of article metadata (see 3.2 Arxiv data collection), this project aims to augment the Hep-Ph citation network in two additional ways; **a)** by enriching each node with an abstract embedding, and **b)** classifying each edge in the network based on the agreement of the citation. This is done for the entirety of the network i.e. we embed 34544 papers (nodes)[1] and label 421567 citations (edges)[2]. These processes result in the new dataset: *Cit-Hep-Ph-Aug*.

4.1.1 Abstract Embeddings

To gain a better understanding of semantic homophily in the Hep-Ph network, we embed all collected abstracts. The goal is to get a dense representation of an abstract that encodes the meaning of the article. These embeddings serve as node attributes in the network, and are then used for clustering groups of articles based on semantic similarity.

To get embeddings, we use current modern LLMs. There are a plethora of models to choose between, some open source, others proprietary. While there are major performance differences between models [24, 25], we are more interested in the procedure of applying LLMs in this context than the actual performance. Naturally, better performing models will yield higher quality embeddings, but as long as the embeddings are of reasonable quality, they suffice for the purposes of this project.

To this end, we use Llama2 [26], an open source LLM created and published by Meta. While it is not the best performing model, it can run locally[3] and is easy to use. Llama2 offers multiple configurations. We use the largest our hardware allows, which is Llama2 13b[4] [27]. The model produces 5120 dimensional embeddings, and we use the model to embed all collected abstracts.

4.1.2 Edge Classification

The most significant part of this paper is adding edge labels. The goal is to expand the original Hep-Ph network with a single label per edge, describing the nature of the relationship. To expand, the objective is adding information regarding whether a paper's citations are in agreement, disagreement or neutral to the papers it cites. The motivation being that adding this information to the network gives a better understanding of it on a structural level, whilst the approach itself can be extended (if proven advantageous) to other similar networks.

[1] There are 34546 papers in the network, however, two were removed for not belonging in the Hep-Ph section (see 3.2 Arxiv data collection).
[2] There are 421578 citations, but 11 of them are connected to the removed papers.
[3] Locally in this case means on IT University of Copenhagen's HPC cluster.
[4] The specific model is `openbuddy-llama2-13b-v11.1.Q5_K_M`.

4.1.2.1 Model Selection

To get the edge labels, we use OpenAI's ChatGPT. Ideally, we would for consistency use the same model for both embedding abstracts and labelling edges. However, due to time constraints, we opt for using ChatGPT over Llama2 for edge labelling. With ChatGPT, since it is a service and not a locally hosted model, we can send multiple prompts to the model concurrently, leading to a significantly higher throughput, only constrained by the API [28] rate limit and pricing.

ChatGPT offers several different models, at different price points and performance. We use `gpt-3.5-turbo`. As of December 2023, it is the cheapest 3.5 model or above, and it handles natural language inference well compared to its predecessors [29].

4.1.2.2 Prompt Engineering

When working with LLM chatbots, prompting is a central challenge. Prompt engineering is still a relatively new concept, and it is widely debated what the best guidelines for maximizing the chatbots' potentials are [20, 21, 22, 23]. However, it has been shown that the quality of the prompt directly influences the quality of the response [19]. In the edge labeling experimentation process, we tried multiple different prompts (see A.2 Prompts) in cascading complexity, to find what works best.

The final prompt we use is the following:

> You are a high energy physics expert. You will get the abstract of a paper that cites another paper within the field of high energy physics. You will evaluate to the best of your ability, whether the paper agrees with the other paper. Paper A cites paper B. The abstract of paper A will follow after "Paper A:", and the abstract of paper B will follow after "Paper B:" You will only provide single word answer, evaluating the agreement between the papers.
> Use "agreement" when Paper A clearly agrees with or builds upon the conclusions of Paper B.
> Use "disagreement" when Paper A clearly contradicts or refutes the conclusions of Paper B.
> Use "neutral" when Paper A is neutral to the conclusions of Paper B or if it is unclear how the papers relate.
>
> Paper A: *Lorem ipsum dolor sit amet...*
> Paper B: *Consectetur adipiscing elit...*

The first part provides context for the model of what its role and task are. It is given specific instructions on what answer to give and in what way. The last part is about when to use which label. This is added in an attempt to make the model more consistent, as it sometimes gives the same edge opposite labels. For maximum reproducibility, we set the seed to 42.

4.1.2.3 Post Collection Cleaning

After all edges are labeled by ChatGPT, we clean the data since the model is not always following the prompts labeling rules. In post processing, we strip and lower all labels. We find all labels containing "disagree", and assign them "disagreement". We find all labels containing "neutral" and label them neutral. Lastly, we find all labels containing "agree" but not "neutral" or "disagree". While this approach is not perfect, we only use it to correct 560 citations. After this correction, 35 citations were left that did not fit any cleaning criteria. These are assigned neutral.

4.2 EDGE LABEL QUALITY ASSESSMENT

Once the dataset has been collected and cleaned, the quality can be evaluated. In the following two sections we will assess the quality of the various generated labels. The first is a quantitative analysis, while the second is a qualitative analysis. For the latter, we brought in a domain expert to provide gold labels for a randomly selected subset of edges.

4.2.1 Quantitative

Getting to the final prompt was an iterative process. To quantify the improvements, we tested multiple prompts, each increasing in complexity and instructions. The experiment setup is applying the same prompt 5 times to 100 citations. This approach enables the assessment of prompt instability. The rationale behind this lies in recognizing the stochastic nature of ChatGPT and the empirical observations that indicate the model's tendency to produce diverse responses for identical queries.

We test 4 different prompts; a baseline, simple, medium, and the final prompt (see appendix section A.2). All of these prompts, are tested on the `gpt-3.5-turbo` model. However, the final prompt undergoes testing on GPT4[5] to uncover the potential of greater models, totalling to 5 experiments.

For each of the 5 prompt/model configurations, we investigate two things: the distribution of labels, and the inter-prompt/model agreement. For the former, we take the mode of each prompt across the 5 runs, and plot the frequencies of labels in a barplot in figure 4a.

For the latter, we use Fleiss' Kappa [30] for evaluating agreement, treating each run of a prompt as an annotator. This does not provide information about how correct the labels are, but it signifies the uncertainty of the prompt/model configuration. If Fleiss' Kappa is low, it means that the configuration yielded many different answers during the 5 test prompts (results seen in figure 4b).

4.2.2 Qualitative

To qualitatively evaluate the quality of the edge labels, we got a PhD student with a masters in high energy physics to label 29 edges. These were all labeled neutral, some with an additional label of connectedness (related, somewhat-related, unrelated), all receiving unrelated or somewhat-related. These 29 edges are a subset of the 100 edges that are used in the quantitative analysis of the label quality. For better insight into ChatGPT's "decision making", we modify our prompt such that it gives a brief explanation of the reasoning behind the label (see **??** Explain prompt - ChatGPT in appendix). While this is not a perfect approach, as the slight altercation to the prompt can alter the models answers, it allows for some interpretation of what it is basing its decisions on, the results of which are shown in results table 2.

4.3 ANALYSIS OF DATA

With the newly augmented Hep-Ph citation network, we demonstrate one way of using the new node attributes (abstract embeddings) and edge labels to improve our understanding of the network. Specifically, we use different partitioning methods, compare them, and inspect their similarity. We employ traditional community detection, multilayer community detection and clustering to partition the data under different assumptions. If these partitions are similar, it can point to the network

[5] It's too expensive to test a lot with this model

topology and/or agreement of citations being shaped by the semantic similarity of papers, and thus be indicative of semantic homophily.

4.3.1 Community Detection

Our analysis begins with community detection, an important step in understanding the underlying structure of a large network. For this purpose we apply six distinct community detection algorithms, namely: Leiden [31], Infomap [32], Fastgreedy [33], Louvain [34], Label Propagation [35], and Walktrap [36]. For each of these methods we report: number of communities, largest community size, and the modularity score. It is presented in table 2 in 5. results.

Our primary method, which also achieves the highest modularity, is Leiden. Leiden is a fairly new algorithm proposed by V. Traag et al [31], as an enhancement of the Louvain algorithm [34]. They propose this algorithm because they discovered, through experimental analysis, that Louvain has a major flaw; it can yield badly connected communities or disconnected communities. They adapt the algorithm, and prove that Louvain's issues are not apparent in Leiden. For this reason, along with computational efficiency, we henceforth focus on this algorithm.

4.3.1.1 Multilayer Community Detection

Having edges labeled with relation types facilitates new ways of analysing the network. The most clear-cut ways of interpreting this new network is as

1. A signed network, where agreement edges are interpreted as $+$, disagreement as $-$, and neutral are either discarded or as $+$.
2. A weighted network with weights $w \in \{1, 0, -1\}$ for agreement, neutral, and disagreement respectively.
3. A multilayer network with an agreement, neutral, and disagreement layer.

In this paper, we focus on the Cit-Hep-Ph-Aug network seen as a multilayer network. In this instance, the network is a special type of multilayer network, since a pair of nodes can only have an edge in one layer at a time. In other words, a citation can only have one relation type at a time. Working with the network as a multilayer network allows for applying multilayer community detection algorithms. The landscape of multilayer community detection is complex [37, 38, 39], and there are many algorithms leading to different partition types. Depending on the desired result, different algorithms and partition types are preferable. While doing a mixed partition type or another more complex partitioning could be interesting, we demonstrate a simple partitioning, where each node belongs to exactly one community, using Leiden.

Multilayer Leiden: For multilayer community detection, we use Leiden [31], which is also the main method we use for standard community detection. Expanding Leiden from single layer to the multilayer case is simple. Instead of maximizing modularity for a single network, it maximizes a weighted modularity across layers [40]. Specifically, with k layers, q_k denoting the quality metric (modularity in this study) in layer k, and w_k being the weight assigned to layer k, the quality of a partitioning is:

$$q = \sum_k w_k q_k$$

The algorithm optimizes this equation in the same way it does it for single layer networks. It introduces additional hyperparameters in the form of the weights w_k for each layer k. With this approach, we can leverage the semantic meaning of each layer in the Cit-Hep-Ph-Aug network. For instance, we can assign the disagreement

layer a negative weight, meaning the algorithm maximises modularity by finding negative community structure in that particular layer. Tuning these weights leads to different partitions. We can assign a higher positive weight to the agreement layer than the neutral, prioritising communities of papers agreeing, while still encouraging neutral citations. To keep this project simple, we only use one configuration of weights to produce the results. The specific configuration is:

$$w = \begin{bmatrix} w_{\mathrm{agreement}} \\ w_{\mathrm{neutral}} \\ w_{\mathrm{disagreemnt}} \end{bmatrix} = \begin{bmatrix} 1 \\ 0.5 \\ -1 \end{bmatrix}$$

4.3.2 Clustering

With abstract embeddings, we can use unsupervised clustering methods to partition groups of articles based on their similarity. There are many possible algorithms for this task; we use K-means for its simplicity. Subsequently, we visualize the clusters using t-distributed stochastic neighbor embedding (T-SNE) for dimensionality reduction.

When using K-means, the number of clusters k is a hyperparameter. We experiment with $k \in \{5, 10, 20, 50, 100\}$ clusters to explore a reasonable search space. For cluster evaluation, we use silhouette score. The silhouette score measures how well-separated the clusters are, providing insight into their cohesion and separation.

4.3.3 Partition Comparison

After using the partitioning techniques, we can compare how they relate in the way they have partitioned the network. We can compare them indirectly by computing modularity and silhouette score for each partition. For the clustering, this means imposing the network topology. For the community detections, we treat the embeddings as node attributes to compute the silhouette score. We also directly compare the partitions using Normalized Mutual Information (NMI), Adjusted Mutual Information (AMI) and Adjusted Rand Index (ARI). Using these metrics, we directly compare the partitions, providing an evaluation of partition similarity.

4.3.4 Homophily with Edge Labels and Node Attributes

As an extension of the partition comparison, we look into the relationship between edge labels and the embedded node attributes, with a focus on studying the presence of homophily. Specifically, we compute the cosine similarity and euclidean distance of embeddings for every edge. Considering we have embeddings with dimension 5120, cosine similarity becomes a powerful choice given it is scale-invariant, meaning it is not affected by the size of the vectors. We compare the average cosine similarity and euclidean distance for the three alignment labels. If the cosine similarity is higher and euclidean distance is lower for agreement edges compared to disagreement or neutral, it could indicate that papers tend to agree with papers that are semantically similar to itself, thus showing homophilic tendencies.

5 RESULTS

5.1 DATASET PRESENTATION

We present "Cit-Hep-Ph-Aug", a citation network for high energy physics (phenomenology) augmented with **a)** an alignment label on each edge, determining if two nodes are in agreement, disagreement or have a neutral relation, and **b)** dense node attributes in the form of an abstract embedding. The distribution of edge labels is seen in figure 2 showcasing ChatGPT 3.5's strong preference for disagreement. The multilayer degree distribution is shown in figure 3. The full dataset can be found on GitHub.

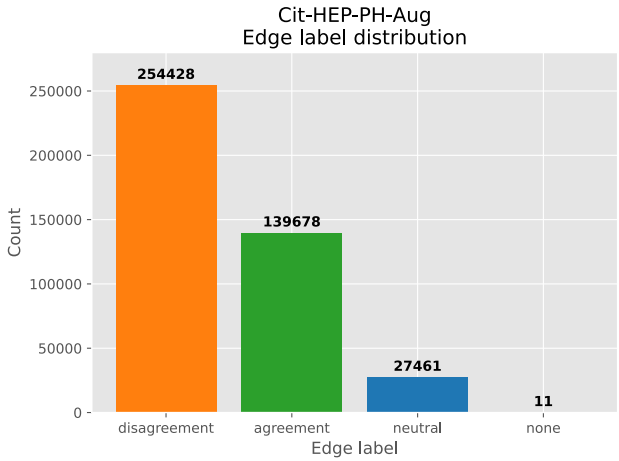
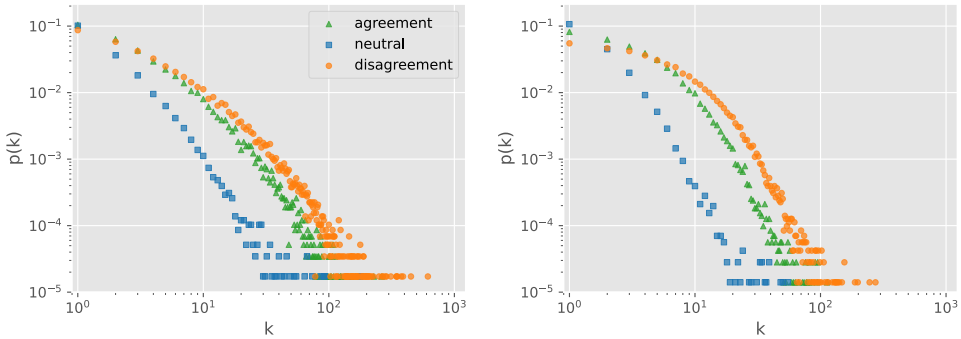


Figure 2: Distribution of edge labels in the Cit-Hep-Ph-Aug network. Note how, contrary to anticipated outcomes, the disagreement label is the most frequent one, followed by agreement.



(a) Cit-Hep-Ph-Aug in-degree distribution for each edge label

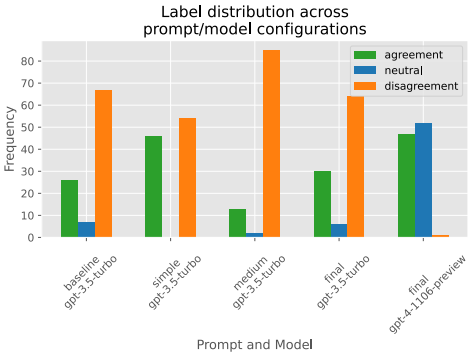
(b) Cit-Hep-Ph-Aug out-degree distribution for each edge label

Figure 3: The in- and out-degree distribution of Cit-Hep-Ph-Aug for each edge label. From the figures, observe how the three labels decline at different rates, with disagreement having the longest tail.

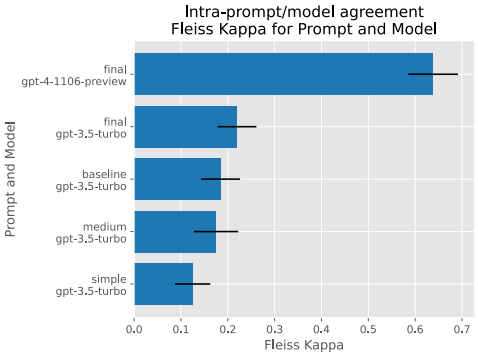
5.2 EDGE LABEL QUALITY

Cit-Hep-Ph-Aug is not perfect. In fact, far from it. Here, we present the results of our quantitative and qualitative approach to understand the imperfections of the data.

In figure 4a, we see the distribution of labels for each prompt/model configuration for the 100 randomly selected citations. The most important takeaway is the difference between ChatGPT 4 and all the other models. ChatGPT 4 uses the disagreement label significantly less, whereas it is the most popular label for all the other configurations. Additionally, as seen in figure 4b, it is also by far the most consistent model in regards to the intra-model agreement, and vastly superior to the other configurations. The configuration we use for collecting the edge labels achieves a Fleiss' kappa of 0.22, with a bootstrapped standard deviation of 0.04. We see that the performance seems to scale with the complexity of the configuration, with the only exception being baseline prompt, scoring higher intra-configuration agreement than medium and simple prompts. However, this result is not conclusive due to uncertainty. It is also important to note that just because the baseline has higher intra-configuration agreement does not mean the labels are more accurate. It strictly means that it more often agrees with itself across multiple runs.



(a) We showcase the resulting labels from the 5 tests. All the GPT3.5 models labels disagreement far more frequently than GPT4.



(b) Plot showing Fleiss' Kappa for the 5 tests. GPT4's answers are considerably more consistent, compared to any of the GPT3.5 variants, showing that ChatGPT 3.5 will often answer differently to the same prompt. Standard deviations are achieved through 1000 bootstraps of the 100 prompts.

Figure 4: Quantitative evaluation of the edge label quality.

Looking at the results through a qualitative lens, table 2 shows the labels given by the expert for 29 edges. Additionally, the expert provided a relation for the two abstracts, alongside one of the semantic labels. This was done for 11 out of the 29 edges, hinting at a deeper understanding of how the papers relate, yet without a larger sample size, it is hard to draw conclusions. Alongside the expert labels, we provide ChatGPT labels, for comparison.

Expanding on the previous results, table 3 shows the Hep-Ph expert labels alongside ChatGPT's reasoning for its labeling, showcasing that they give widely different results, even if ChatGPT at times seems convincing. The full dataset can be found on GitHub.

Table 2: Aggregate statistics of qualitative results by field expert and ChatGPT 3.5. The sample size is very small, but with 29 labeled neutral edges by the expert out of 29 samples, it indicates that the concept of agreement might be ill-posed.

Category	Expert	ChatGPT
Agreement	0	16
Neutral	29	6
Disagreement	0	7
Related	2	-
Somewhat-related	7	-
Unrelated	2	-

Table 3: Examples of qualitative assessments. We have the expert's label & relation, alongside ChatGPT's label and an explanation of how ChatGPT made its decision. To get an explanation from the model, we created a prompt to explain why ChatGPT chose the way it did. This helps us understand how it makes its decision. The full dataset can be found on GitHub.

id_a	id_b	ChatGPT Label	Expert Label	Expert Relation	ChatGPT Explanation
207372	9508264	disagreement	neutral	unrelated	Explanation: Paper A focuses on SU(3)-flavor symmetry breaking and quark-antiquark annihilation in describing singlet-octet mixing for several nonets, using the mass matrix formalism. In contrast, Paper B discusses the decay widths of specific $s\bar{s}$ mesons, such as 1^3F_2 and 1^3F_4 (related to $\xi(2220)$), and proposes interpretations involving different hadronic states at the same mass. The different emphasis and content indicate a disagreement in their primary objectives and conclusions
9903329	9706239	agreement	neutral	related	Paper A considers day-night asymmetries predicted by MSW solutions of the solar neutrino problem and discusses the incoherent nature of the ν_e-regeneration effect inside the Earth. Paper B calculates the number of solar neutrino events with and without neutrino oscillations, evaluating distortions predicted by the MSW effect in the zenith angle distributions of solar neutrino events. Both papers focus on MSW solutions and discuss the regeneration effect of ν_e from ν_μ or ν_τ due to $\nu - e$ interactions in the Earth. Therefore, Paper A agrees with and builds upon the conclusions of Paper B, as both contribute to the understanding of MSW solutions and their impact on solar neutrino events.
201032	9308339	neutral	neutral	related	Explanation: Paper A and Paper B both explore the production of new heavy particles in the context of extensions to the Standard Model. However, Paper A focuses on the electron-proton collider for the Very Large Hadron Collider at Fermilab, specifically studying the reaction $e^-p \longrightarrow e^+ + jets$. On the other hand, Paper B analyzes the production of new heavy fermions in e^+e^- collisions. While both papers delve into the realm of new heavy particles, they address different collider scenarios and production mechanisms. The nature of their investigations is distinct, making it challenging to definitively categorize the relationship as agreement or disagreement. Therefore, the assessment is neutral
9907239	9601369	disagreement	neutral	*No relation provided by the expert*	The topics of Paper A and Paper B are unrelated. Paper A discusses the formation of observable physical space and the existence of global anisotropy based on a model involving discrete vectorial objects. On the other hand, Paper B focuses on investigating weak transition form factors among heavy pseudoscalar mesons within a relativistic quark model. There is no apparent connection or agreement between the conclusions of the two papers.

5.3 ANALYSIS OF DATASET

For the analysis of the dataset, we present 5 figures/tables. The first is the performance of multiple community detection algorithms on the largest weakly connected component of the Hep-Ph network, as seen in table 4. The algorithm achieving the highest modularity is Leiden, which is also the algorithm used for the rest of the analysis. It provides the lowest number of communities, while maximizing modularity.

Table 4: Results of various community detection algorithms run on the largest weakly connected component of Hep-Ph citation network. Running these algorithms multiple times will yield slightly different results due to the stochastic nature of the processes. All algorithms are run with default parameters using their igraph implementation in Python, except for Leiden, which uses Leidenalg.

Method	Number of Communities	Largest Community Size	Modularity
Louvain	20	5171	0.72
Leiden	24	4059	0.73
Infomap	792	1635	0.61
Fastgreedy	131	12568	0.55
Label Propagation	273	5330	0.68
Community Walktrap	678	7000	0.68

The second result is table 5 showing each partition type evaluated with silhouette score, modularity, and modularity in the individual layers of the multilayer interpretation. It conforms to our intuition. Leiden scores the highest modularity, with multilayer community detection as a runner up. The reason for multilayer community detection performing worse than basic community detection is due to its negative weight on the disagreement layer. This is reflected in the disagreement layers modularity, where multilayer community detection achieves 0.1830 as opposed to the standard community detection's 0.7171. This is expected, since the multilayer method minimizes disagreement modularity through its layer weights.

Interestingly, k-means clustering only scores a maximum silhouette score of 0.0402, for $k = 5$. There are a multitude of potential factors for this, some of which are discussed later.

The most interesting observation in the table is multilayer community detection's silhouette score of -0.2985. This is remarkably low, especially considering the results of table 7, where we see no real link between edge type and embedding space similarity.

Comparing groupings with NMI, AMI, and ARI, we see that unsurprisingly, the most similar groups are community detection and multilayer community detection getting 0.35, 0.30 and 0.12 on the metrics respectively. The second most similar partitioning is clustering with $k = 100$ and multilayer community detection.

Table 5: Overview of the 3 partition types and their associated metrics. The algorithms used are k-means, Leiden, and multiplex leiden. The modularity is first presented as a full score, and subsequently for each layer. We observe an incredibly low Silhouette score across all the partitions, even for clustering. Meanwhile, the modularity behaves as expected with community detection having the highest, followed by the multilayer variant. All partitions are run with seed 42 for reproducibility. Hyperparameter configurations are listed below the partions name. The rest of the table can be found in Appendix 8.

Partition type	**Silhouette**	**Modularity**	**Agreement** **Neutral** **Disagreement**
Clustering $k = 5$	0.0402	0.0223	0.0298 0.0224 0.0177
Community detection Leiden	−0.0849	0.7331	0.7723 0.6753 0.7172
Multilayer community detection $w_{agr} = 1\ w_{neu} = 0.5\ w_{dis} = -1$	−0.2985	0.3259	0.5200 0.6624 0.1830

Table 6: Comparison of partitioning methods, containing NMI, AMI, and ARI for all combinations for methods.

k	**Partition type 1**	**Partition type 2**	**NMI**	**AMI**	**ARI**
	Community detection	Multilayer community detection	0.35	0.30	0.12
5	Community detection	Clustering	0.006	0.005	0.002
	Clustering	Multilayer community detection	0.049	0.004	0.0006
10	Community detection	Clustering	0.022	0.021	0.010
	Clustering	Multilayer community detection	0.071	0.012	0.003
20	Community detection	Clustering	0.030	0.0281	0.012
	Clustering	Multilayer community detection	0.091	0.019	0.004
50	Community detection	Clustering	0.036	0.032	0.008
	Clustering	Multilayer community detection	0.111	0.207	0.004
100	Community detection	Clustering	0.05	0.04	0.005
	Clustering	Multilayer community detection	0.1357	0.285	0.004

Table 7: Overview of average euclidean distance and cosine similarity based on edge type. Contrary to expectations, there is no significant difference in the embedding space between agreement, neutral, or disagreement edges.

	Average Euclidian Distance	**Average Cosine Similarity**
Agreement	34.4	0.88
Neutral	35.94	0.87
Disagreement	34.88	0.88

The results seen in table 7 indicate minimal variations in both Euclidean distance and cosine similarity among agreement, neutral, and disagreement edges. These lackluster results are most likely a combination of two shortcomings of our work: The Llama2 generated embeddings and the GPT 3.5 edge labeling. The embedding space is ill-defined, most likely due to the "simplicity" of the model and the difficult domain it was deployed in. Furthermore, we showcase above how that the labeling process have room for substantial improvement.

Lastly, we reduce the dimensionality of the article embeddings with t-SNE to 2 dimensions to visualize it in a scatterplot. The plot itself does not show much, and it does not seperate the embedding space into clusters particularly well, despite being optimized for it. This could either be due to the dimensionality of the embeddings or their quality.

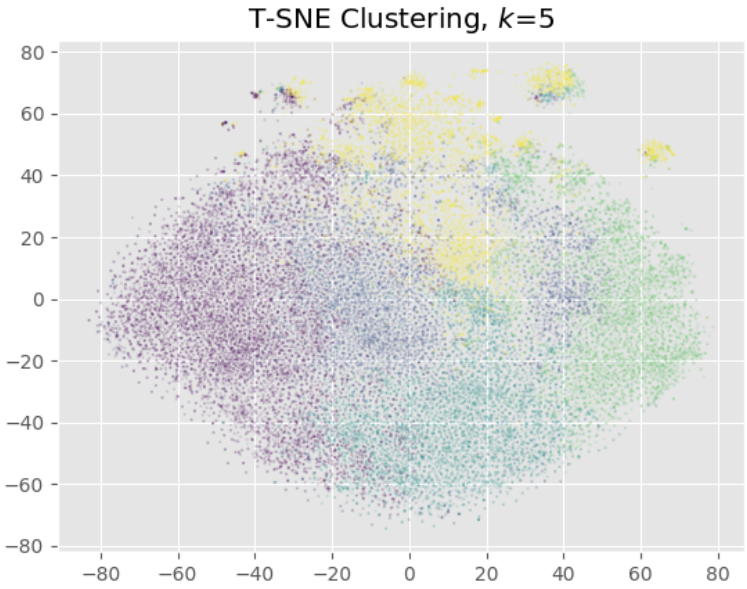


Figure 5: Scatterplot of dimensionality reduced abstract embeddings, colored with K-means clustering with $k = 5$.

6 ANALYSIS

The created Cit-Hep-Ph-aug dataset has some major flaws that prevent the robustness of any analysis. While the approach and intuition is sound, the specifics of the implementation and the problem of what "agreement" means limits the usefulness.

6.1 PROBLEMS OF AGREEMENT/NEUTRAL/DISAGREEMENT

Contradicting citations have been shown to be very uncommon [41], even ones citing papers that have been retracted, suggesting that the disagreement label should be a tiny amount of the total labels. While ChatGPT 4 succeeds in this regard, our ChatGPT 3.5 labels fail. The paper behind scite.ai [17] also highlights the problem of what counts as a contrasting citation, limiting it to instances where new evidence is presented, but noting that it could make sense to include conceptual and logical arguments against a paper. They also recognize that their model will sometimes be wrong, and allow for users to flag, and suggest a new classification. These will then "be reviewed independently by two experts and, if both agree, will be accepted and labeled as 'Expert classified'" [42]. In our case we define agree and disagree as the following: We use agreement when a paper clearly agree with or builds upon another paper. We use disagreement when a paper clearly contradicts or refutes the conclusion of another paper. The question then becomes this: if its neither of these two options, should it then just automatically become a neutral? Our expert introduced an interesting notion for this problem. He started to label their relation as well, and not only their agreement level.

6.1.1 Qualitative Analysis

The expert initially labeled the 29 papers as neutral but during the process of labelling he decided to include labels indicating whether the papers are related. Table 3 shows some selected examples of the edges that both the expert labeled and ChatGPT gave an explanation for. The 29 papers were also fed to ChatGPT with a special prompt requiring it to explain its thought process behind each label. This is an important factor if we wish to get an understanding of its thought process. ChatGPT's response is a black box, meaning that we will never be completely certain as to how it generates it's output, or where the information comes from. In hindsight there are possible improvements: if the choice of model for labeling is ChatGPT 3.5, the custom instruction prompt should be more specific.

6.1.2 Quantitative Analysis

Understanding connections between papers isn't always clear from just the abstract. It also seems to be unclear to the expert what criteria should be met for the label to be agree or disagree. ChatGPT 3.5 tended to label both agreement and disagreement more frequently than neutral, but the explanations it provided lacked consistency. It sometimes labeled papers as neutral or in agreement, when they were related but not in disagreement. On the other hand, it labeled disagreement for papers with "no apparent connection". While some explanations made sense, the conflicting conclusions suggest that results could improve by using a better model or using the entire article instead of the abstract. In hindsight, it might have been better to not

only label the edge but also provide a relation to it. This would encode additional information about the connection between two papers. All the papers labeled by the expert were neutral, but they differed in their relation to each other. If ChatGPT could provide a relation alongside the agreement label, it would offer a deeper understanding of the edges and add more substance to the multilayer analysis.

6.1.3 Ideal Balance of Labels

Comparing to Kumar's paper [10] presented in the 2 background, we see that his findings contradicts ours. In his paper, he presents that the neutral label is the most common, and the disagreement label is the most rare. This seems to correspond with what our expert labeled, although his labeled edges is a very small sample size. By looking at figure 2 it is clear that GPT 3.5 didn't make the same conclusions as our expert or Kumar. This most likely boils down to the model simply not being good enough. If we look at figure 4a, we see from our test run of GPT 4, that it gets significantly better results.[1]

6.2 IMPROVING LLMS

A major bottleneck regarding the homophily analysis is the abstract embeddings. While sound in principle, they do not show much in practise. This is evident in the clustering, where we find close to no discernible patterns of similarity or cohesive groupings. The most obvious reasons for this is the model used for embeddings. We use Llama2 13B, specifically `openbuddy-llama2-13b-v11.1.Q5_K_M`. The model itself is not the best performing model, even for its parameter category. For instance, Mistral 7B have been shown to outperform Llama 13B on a range of LLM benchmarks, despite its smaller size [43]. In retrospect, this model might have yielded better results.

Another issue with LLama2 is, it is not fine-tuned to the domain. High energy physics as a field uses a lot of domain specific language, and without a basic understanding of the field, it can be hard even for humans to understand. This could explain why we see no real spread in the embeddings. The model does not understand the domain well enough to distinguish between the intricacies of the field. Fine-tuning a model could, at least partially, solve this issue.

Similarly, the edge labels suffer from the same limitations. While ChatGPT 3.5 has been shown to outperform Llama2 13B, our results indicate that it is not good enough for the task. ChatGPT 4 performs seemingly good, however, the pricing of the model makes it infeasible to use it. Instead, using a fine-tuned model on the Hep-Ph field could lead to higher quality edge labels.

These issues are also shown in table 7. The table shows average cosine similarity and euclidean distance between to nodes based on edge type. If everything followed expectations, we would see agreement labels leading to more similarity in the abstract embeddings, and more dissimilarity in the disagreement labels. However, they are all practically equivalent. One plausible reason is the that the Llama2 model has a hard time distinguishing between intricacies of the Hep-Ph field.

[1] Provided better is getting the same labels as Kumar and our expert

7 CONCLUSION

In this paper, we show an approach to applying LLMs for augmenting networks with text attributes. Specifically, we demonstrate one such approach on the Hep-Ph citation network, creating an augmented version, Cit-Hep-Ph-Aug. We successfully embed all abstracts in the Hep-Ph data using Llama2 13B, and manage to label all edges in the network based on the semantic relation of a citation with labels "agreement", "neutral" or "disagreement" using ChatGPT 3.5. While the implementation specifics of the approach are debatable, and the result severely lacking, our findings indicate that the approach itself is sound, and provided more research, has the potential to lead to more ways of analysing networks.

Additionally, we conduct an analysis of Cit-Hep-Ph-Aug, applying clustering to abstract embeddings, community detection on the Hep-Ph network, and multilayer community detection on the Cit-Hep-Ph-Aug data, interpreting the network as a 3 layer network with an agreement-, neutral- and disagreement layer. The goal was to analyse the influence of homophily on the network topology, yet we do not manage to show any indication of it, likely due to the quality of the gathered data.

With this study, there are two major limitations. The first is the applied LLMs. The LLama2 13B embeddings are lacking in quality, likely due to the difficulty of the domain and the capabilities of the model. Using other models and fine tuning would likely yield improvements. Similarly, ChatGPT 3.5 performs significantly worse compared to its GPT 4 counterpart. Secondly, the notion of agreement, neutrality and disagreement is debatable, with other studies and our own investigations showing that disagreement is a rare phenomena in this field. Instead, it would be beneficial to revamp the labeling goal and find another more well defined target label.

BIBLIOGRAPHY

[1] Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. Gpts are gpts: An early look at the labor market impact potential of large language models, 2023.

[2] Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103:102274, 2023.

[3] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

[4] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, 2021.

[5] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 2022.

[6] Microsoft Research AI4Science and Microsoft Azure Quantum. The impact of large language models on scientific discovery: a preliminary study using gpt-4, 2023.

[7] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187, 2005.

[8] Dangzhi Zhao and Andreas Strotmann. *Analysis and visualization of citation networks*. Morgan & Claypool Publishers, 2015.

[9] Sune Lehmann, Benny Lautrup, and Andrew D Jackson. Citation networks in high energy physics. *Physical Review E*, 68(2):026113, 2003.

[10] Srijan Kumar. Structure and dynamics of signed citation networks. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, page 63–64, Republic and Canton of Geneva, CHE, 2016. International World Wide Web Conferences Steering Committee.

[11] Johannes Gehrke, Paul Ginsparg, and Jon Kleinberg. Overview of the 2003 kdd cup. *Acm Sigkdd Explorations Newsletter*, 5(2):149–151, 2003.

[12] Michele Coscia. The atlas for the aspiring network scientist. *CoRR*, abs/2101.00863, 2021.

[13] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. Statistical properties of community structure in large social and information networks. In *Proceedings of the 17th international conference on World Wide Web*, pages 695–704, 2008.

[14] Arun S Maiya and Tanya Y Berger-Wolf. Sampling community structure. In *Proceedings of the 19th international conference on World wide web*, pages 701–710, 2010.

[15] Lars Backstrom and Jure Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 635–644, 2011.

[16] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354, Vancouver, British Columbia, Canada, October 2005. Association for Computational Linguistics.

[17] Josh M. Nicholson, Milo Mordaunt, Patrice Lopez, Ashish Uppala, Domenic Rosati, Neves P. Rodrigues, Peter Grabitz, and Sean C. Rife. scite: A smart citation index that displays the context of citations and classifies their intent using deep learning. *Quantitative Science Studies*, 2(3):882–898, 11 2021.

[18] Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. Sentiment analysis in the era of large language models: A reality check, 2023.

[19] Amarachi B. Mbakwe, Ismini Lourentzou, Leo Anthony Celi, Oren J. Mechanic, and Alon Dagan. Chatgpt passing usmle shines a spotlight on the flaws of medical education. *PLOS Digital Health*, 2(2):1–3, 02 2023.

[20] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt, 2023.

[21] Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution, 2023.

[22] Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. Sentiment analysis in the era of large language models: A reality check, 2023.

[23] Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. Large language models understand and can be enhanced by emotional stimuli, 2023.

[24] Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. Open llm leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard, 2023.

[25] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.

[26] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao,

Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[27] Hardwarecornr. Llama-2 llm: Versions, prompt templates hardware requirements, 2023. Last visited on 13 December 2023.

[28] OpenAI. Openai platform documentation, 2023. https://platform.openai.com/docs/overview (Last accessed on December 15, 2023).

[29] Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, Jie Zhou, Siming Chen, Tao Gui, Qi Zhang, and Xuanjing Huang. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models, 2023.

[30] J. L. Fleiss. *Measuring nominal scale agreement among many raters*. Psychological Bulletin, 1971.

[31] V. A. Traag, L. Waltman, and N. J. van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1), March 2019.

[32] Martin Rosvall and Carl T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, January 2008.

[33] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70:066111, Dec 2004.

[34] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, October 2008.

[35] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3), September 2007.

[36] Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks (long version), 2005.

[37] Obaida Hanteer, Roberto Interdonato, Matteo Magnani, Andrea Tagarelli, and Luca Rossi. Community detection in multiplex networks. *CoRR*, abs/1910.07646, 2019.

[38] Jungeun Kim and Jae-Gil Lee. Community detection in multi-layer graphs: A survey. *ACM SIGMOD Record*, 44(3):37–48, 2015.

[39] Xinyu Huang, Dongming Chen, Tao Ren, and Dongqi Wang. A survey of community detection methods in multilayer networks. *Data Mining and Knowledge Discovery*, 35:1–45, 2021.

[40] leidenalg documentation. https://leidenalg.readthedocs.io/. [Accessed 14-12-2023].

[41] Frédérique Bordignon. Self-correction of science: a comparative study of negative citations and post-publication peer review. *Scientometrics*, 124, 06 2020.

[42] scite. How are citations classified?, 2023.

[43] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

A APPENDIX

A.1 SELF-LOOPS

List of nodes with self-loops - the numbers are their Arxiv ID:

- 9803242
- 6056
- 108195
- 9705442
- 9807361
- 102122
- 9707481
- 9701288
- 9610527
- 9907261
- 201248
- 9607208
- 9612326
- 9502380
- 9409427
- 2193
- 9809337
- 9405254
- 9311294
- 201291
- 9312343
- 5253
- 9607359
- 9605228
- 9610448
- 106319
- 111193
- 107254
- 103101
- 7349
- 203041
- 9807206
- 212116
- 9905499
- 9612311
- 9311351
- 9702366
- 212229
- 206192
- 302248
- 210046
- 9309327
- 110207
- 9903271

A.2 PROMPTS

Various prompts were tested during the development of this project. Below, they are listed, along with the label we assigned them.

- Baseline: You are given two abstracts; one from Paper A and one from Paper B. Paper A cites Paper B. Give a single word answer, evaluating the agreement between the papers. Use one of these labels: agreement, disagreement, neutral.
- Simple: You are a high energy physics expert. You will get the abstract of a paper that cites another paper within the field of high energy physics. You will evaluate to the best of your ability, whether the paper agrees with the other paper. Paper A cites paper B. The abstract of paper A will follow after "Paper A:", and the abstract of paper B will follow after "Paper B:" You will only provide single word answer, evaluating the agreement between the papers.
- Medium: You are a high energy physics expert. You will get the abstract of a paper that cites another paper within the field of high energy physics. You will evaluate to the best of your ability, whether the paper agrees with the other paper. Paper A cites paper B. The abstract of paper A will follow after "Paper A:", and the abstract of paper B will follow after "Paper B:" You will only provide single word answer, evaluating the agreement between the papers. Use "agreement" when Paper A agrees with Paper B. Use "disagreement" when Paper A contradicts Paper B. Use "neutral" when it is neither "agreement" or "disagreement".
- Final: You are a high energy physics expert. You will get the abstract of a paper that cites another paper within the field of high energy physics. You will evaluate to the best of your ability, whether the paper agrees with the other paper. Paper A cites paper B. The abstract of paper A will follow after "Paper A:", and the abstract of paper B will follow after "Paper B:" You will

only provide single word answer, evaluating the agreement between the papers. Use "agreement" when Paper A clearly agrees with or builds upon the conclusions of Paper B. Use "disagreement" when Paper A clearly contradicts or refutes the conclusions of Paper B. Use "neutral" when Paper A is neutral to the conclusions of Paper B or if it is unclear how the papers relate. """,

- Explain: You are a high energy physics expert. You will get the abstract of a paper that cites another paper within the field of high energy physics. You will evaluate to the best of your ability, whether the paper agrees with the other paper. Paper A cites paper B. The abstract of paper A will follow after "Paper A:", and the abstract of paper B will follow after "Paper B:" You will only provide single word answer, evaluating the agreement between the papers, followed by an explanation of your reasoning. Use "agreement" when Paper A clearly agrees with or builds upon the conclusions of Paper B. Use "disagreement" when Paper A clearly contradicts or refutes the conclusions of Paper B. Use "neutral" when Paper A is neutral to the conclusions of Paper B or if it is unclear how the papers relate.

A.3 PARTITION METRICS

Table 8: Additional partition metrics for clustering with k > 5, which is in table 5

Partition type	Silhouette	Modularity	Agreement Neutral Disagreement
Clustering (k=10)	0.0235	0.0338	0.0405 0.0294 0.0299
Clustering (k=20)	0.0206	0.0285	0.0357 0.024 0.0245
Clustering (k=50)	0.0128	0.0178	0.0228 0.0151 0.0151
Clustering (k=100)	0.0097	0.0135	0.0168 0.0114 0.0118