Enhancing Gene Set Overrepresentation Analysis with Large Language Models

Authors: Jiqing Zhu[1], Rebecca Y. Wang[1], Xiaoting Wang[1], Ricardo Azevedo[1], Alexander Moreno[1], Julia A. Kuhn[1], Zia Khan[1,+]

Affiliations: 1. Alector, Inc.
+ corresponding author: Zia Khan zia.khan@alector.com

Abstract

Gene set overrepresentation analysis is a widely used approach for interpreting high-throughput transcriptomics and proteomics data. However, traditional methods rely on static, human-curated gene set databases that limit flexibility. We introduce *llm2geneset*, a framework that leverages large language models (LLMs) to dynamically generate gene sets from natural language descriptions. Through benchmarking against curated gene set databases, we show that LLM-generated gene sets are significantly overrepresented in corresponding human-curated gene sets. Additionally, LLMs can propose multiple biological processes for input sets of differentially expressed genes (DEGs), enabling the identification of overlapping pathways. Applying *llm2geneset* to RNA-seq data from iPSC-derived microglia treated with a *TREM2* agonist reveals more interpretable and relevant pathways compared to static databases, demonstrating the potential of LLMs for flexible, context-aware gene set generation. This approach enhances hypothesis generation and improves the interpretation of high-throughput biological data. *llm2geneset* is available as open source at: https://github.com/Alector-BIO/llm2geneset.

Introduction

Gene sets are powerful tools for hypothesis generation from transcriptomics and proteomics data sets. Gene sets group genes into biological processes, molecular functions, cellular components, and pathways [1,2]. They are often more interpretable biological concepts than individual genes. Many different classes of statistical tests can be used to determine whether gene sets are significantly enriched among differentially expressed genes (DEGs) (or differentially abundant proteins) – the simplest rely on Fisher's exact test or the hypergeometric distribution [2]. This is a powerful approach because it provides a means to go from a list of genes obtained by differential expression studies to more interpretable hypotheses regarding underlying biology impacted in an experiment.

While statistical tests for gene set enrichment have been well studied, nearly all of these tests rely on external, human-curated gene set databases [3–7]. These databases must be regularly updated to capture advancing scientific knowledge and are often challenging to tailor to specific questions and studies. Choice of gene sets and gene set databases is guided by questions of interest to the study - or simply by convenience - and can greatly impact the interpretation and hypotheses derived from transcriptomics studies [1]. In this paper, we demonstrate that LLMs offer an alternative to these mostly static databases –

allowing gene sets to be generated based on natural language. This reduces the cost associated with curating and maintaining gene sets, enables gene sets to be customized more easily, and leverages the reasoning capability of LLMs. When combined with overrepresentation analysis, we show that this approach enhances the ability of LLMs to discover biological processes enriched in a list of genes.

LLMs have recently found several uses in the analysis of transcriptomics data sets and their application to problems in computational biology is an active area of research. LLMs have been used to automate cell type annotation in single cell data sets on the basis of their ability to capture marker genes of known cell types [8]. Recent work has demonstrated that LLMs can discern functional relationships within a given set of genes, yet the statistical relevance of these LLM derived observations has not been considered [9,10]. LLMs have been augmented with bioinformatic specific tools to solve tasks [11]. Yet, use and evaluation of LLMs as a tool for flexible gene set generation and overrepresentation analyses has not been explored.

In this study, we evaluate several LLM prompting strategies to generate gene sets based on natural language descriptions of these gene sets. Given that there is no agreed upon definition of a ground truth gene set, we propose an evaluation strategy where we examine the overrepresentation gene sets generated by LLM in the gene set obtained by a human curator with the same description. We observe statistical overrepresentation for a large fraction of these gene sets that improves with LLM capability - indicating that LLMs capture many relationships between biological concepts and sets of genes. We find that the precision of this relationship can be improved by use of prompt ensembling and by eliciting model confidence. We demonstrate that LLMs can dynamically propose and generate gene sets and leverage overrepresentation analysis to determine the biological processes present in a set of DEGs. This is particularly true in contexts where multiple biological processes are represented by this set of DEGs. By providing a flexible and steerable framework for gene set generation, we illustrate that our approach improves the interpretability of differential expression studies.

Results

LLMs Generate Informative Gene Sets

We hypothesized that LLMs could be used to generate gene sets using flexible natural language descriptions of these gene sets. To achieve this, we used a fixed text prompt with a field that can be replaced by the natural language description of the gene set (see Methods) [12,13]. Our prompt consisted of several elements: (1) a role-prompt, a strategy whereby a system message is used to guide the generations of an LLM; (2) the prompt itself requesting generation of genes for a given biological process or cellular component; (3) a formatting component to guide the LLM to produce output uses standard HUGO Gene Nomenclature Committee (HGNC) gene symbols and that could be programmatically loaded into data structures for downstream use in gene set tests (Fig. 1A). To evaluate the

output of an LLM, we assessed whether the gene sets generated by an LLM were significantly overrepresented in corresponding human curated gene sets (Fig. 1B). To compute whether the overrepresentation we observed was not due to chance, we used the hypergeometric distribution to compute a p-value for a one-tailed Fisher's exact test and adjusted for multiple testing (see Methods).

We performed this assessment using natural language descriptions of gene sets in 3 human curated gene set databases: KEGG [14], Reactome [15], and Wikipathways [16]. Curated for different applications, each gene set database captured differing underlying biology and varied in total number of gene sets (Fig. S1A). In total, we examined overrepresentation in 2,939 gene sets and the gene sets varied in size ranging from a median of 24-78 genes (Fig. S1B). We evaluated gene set generation using 3 different language models: GPT-3.5, GPT-4o-mini, and GPT-4o [17,18] (see Methods). We found that between 66% to 94% of gene sets in KEGG, Reactome, and Wikipathways were significantly overrepresented in LLM generated gene sets at a Bonferroni adjusted p-value of 0.01 (Fig. 1C). We observed a higher fraction of overrepresented gene sets generated by the more capable GPT-4o model than GPT-3.5. This fraction tracked with the relative capabilities of each model we considered with GPT-4o mini ranking between GPT-4o and GPT-3.5. Across most gene sets, the over-representation p-value was smallest for GPT-4o indicating that on an individual gene set level a more capable model yields gene sets in greater agreement with curated gene sets (Fig. 1D). Our results illustrate that LLMs, when given a natural language description of a gene set, can accurately generate the genes associated with the biological concept in this description.

The prompt we evaluated above relied on two additional components: a role prompt and a format specification to generate output that be used for downstream applications. Role prompting is a strategy by which to shape the output of an LLM by asking it to take on a specific role (see Methods) [13]. We evaluated whether the role-prompt had any impact on the quality of the results. We found that the role-prompt had little impact on the evidence for overrepresentation of curated genes in LLM generated gene sets (Fig. S1C). Formatting is crucial for subsequent use of the gene sets generated by LLMs. We assessed how often each LLM used HGCN symbols, as directed in the prompt, by comparison to gene symbols in Ensembl (GRch38.p14, release 112). We found that on average 95% of the gene symbols returned by GPT-4o were HGCN gene symbols (Fig. S1D). This number was 88% for GPT-3.5 and GPT-4o mini. We additionally noticed a tendency of the models to return duplicate gene symbols for a given gene set description. We found that GPT-3.5 generated gene sets with one or more duplicate gene symbols across 13-24% of gene sets, which was greatly reduced for GPT-4o (5-6%) (Fig. S1E). Overall, the LLMs we evaluated followed formatting instructions enabling generation of HGCN gene symbols for downstream use.

Token use is a crucial metric for LLMs and directly tied to compute cost [19]. Input tokens and output tokens are considered differently given that input tokens can be cached to reduce compute cost [20]. We quantified the number of input and output tokens used to generate LLM version of each database and model (Fig. S1F). Given that GPT-4o and GPT-

4o mini both use the same tokenizer, 528,666 tokens were used as input the gene set descriptions from KEGG, Reactome, and Wikipathways. We found that GPT-4o used the most tokens in its output across databases. 750,278 output tokens were required to generate 2,939 gene sets using GPT-4o. In contrast, GPT-4o mini required 667,503 output tokens. Quantification of token usage indicates that the compute cost associated with gene set generation was minimal overall.

Ensembling and Confidence Improve the Precision of Gene Set Generation

LLM prompting is an active area of research [13]. On question-and-answer (Q&A) benchmarking data sets various prompting strategies have been explored to increase LLM performance. We evaluated 3 prompting strategies on the problem of gene set generation: model reasoning, model confidence, and ensembling (Fig. 2A, see Methods). Encouraging an LLM to provide reasoning for why and how it arrived at a particular answer has been demonstrated to be a powerful approach to improve performance on Q&A datasets [21]. We modified our prompt to require that the model to provide a single sentence for its rationale for why a gene was included in a gene set (see Methods). We additionally considered whether we could elicit the model's confidence in whether it thought a gene belonged to a gene set. We required the model to indicate low, medium, and high confidence for each gene, limiting gene sets to high confidence genes. We also considered ensembling. We used multiple generations of the model with differing random seed values. Across generations, we identified genes that the model consistently associated with a given gene set description. When a gene is consistently provided as output for a query for a gene set description, this may represent fixed and highly confident knowledge that the LLM captures.

For each of these approaches reasoning, confidence, and ensembling, we observed a small decrease in the number of LLM generated gene sets that were overrepresented in human curated gene sets (Fig. 2B). The impact was more modest for GPT-4o. As expected, all of the gene sets generated by ensembling were the same size or smaller than those generated by our original prompt. We hypothesized that genes that appeared consistently in model generations with different seeds reflected the model's higher confidence the gene belonged to a given gene set. To quantify this, we evaluated whether these gene sets were more precise. Precision can be quantified by the number of genes in the LLM generated gene set belong to a curated gene set over the total number of genes returned. We found that both ensembling and model confidence, but not model reasoning significantly improved the precision of gene sets returned (Fig. 2C). Each of the strategies we considered requires additional tokens, thus increasing cost. As expected, token usage was highest for the ensembling strategy followed by model reasoning (Fig. S2A-S2B). Given the considerably higher token usage of ensembling, our experiments suggest that model self-reported confidence is the better approach to increase the precision of gene sets with minimal additional cost in additional tokens. Taken together, our results illustrate that prompting strategies can tune the precision of gene sets generated by LLMs.

LLM Generated Gene Sets Enable Discovery of Multiple Enriched Biological Processes

LLMs have recently been proposed as tools to discover biological processes represented by an input set of genes [10]. In this setting, an LLM prompt includes a set of genes and asks an LLM to generate natural language description of the biological process these genes participate in – the reverse of our evaluation approach above. The prompt is evaluated based on how closely the LLM can recover the ground truth natural language description of the gene set in the output. As the LLM may not use the same exact description as a curator used for a gene set, agreement can be quantified by, for example, how many words (unigrams) or word pairs (bigrams) from the ground truth description are present in the output of the LLM gene description. Agreement can also be quantified by use of text embedding models and computation of cosine similarity between an embedding of the LLM description and an embedding of the ground truth description for a gene set [22].

To extend our approach to map genes to their biological processes, we used a second "proposal" prompt that uses an LLM to propose potential biological processes and cellular components given a set of genes (see Methods, Fig. 3A). Using these proposed biological processes, we generated gene sets for each of these processes and evaluated whether we observed significant overrepresentation of these gene sets within the genes under consideration. Crucially, the generation of the gene sets can only use the description of the gene set and no other model context (see Methods). This approach, which we call *llm2geneset*, is equivalent to generating a small custom gene set database and assessing whether these custom gene sets in that database are overrepresented in the gene set for which we are trying to uncover function.

We compared our approach to a prompting strategy called GSAI [9,10]. In contrast to our approach, GSAI used a single prompt to request the LLM return the biological process the input genes belong to along with both reasoning and confidence. Using benchmarking gene sets from KEGG, Wikipathways, and Reactome, we found that *llm2geneset* generated gene set descriptions had on average a higher fraction of shared unigrams and bigrams as well as higher cosine similarity with the ground truth gene set descriptions than GSAI (Fig. 3B, S3A). In terms of computational cost, our approach required a comparable number of input tokens to GSAI, but a larger number of output tokens (Fig. S3B). This comparison suggests that combining gene set generation with our proposal approach can improve the discovery of the biological process represented by a given gene set.

Experiments often influence multiple biological processes, leading to the discovery of distinct overrepresented gene sets that include DEGs. To simulate such a scenario, we selected gene sets where the biological processes were recovered by both the GSAI prompt and *llm2geneset* with high similarity to the ground truth gene set description (>0.7 cosine similarity) for each benchmarking gene set database. Next, we combined 50 pairs of these gene sets from each gene set database. We then evaluated the recovery of the distinct biological processes from these mixed gene sets, comparing the GSAI prompt with our approach across LLMs (see Methods). We found that with the GSAI prompt LLMs were

frequently unable to identify multiple gene functions in a gene set. In contrast, our approach more frequently recovered multiple functions as reflected by higher fraction of shared unigrams and bigrams as well as cosine similarity (Fig. 3C, Fig. S3C). Our analysis indicates that coupling generation of gene sets by LLM to overrepresentation analysis is a more effective strategy than model prompting to uncover multiple distinct gene set functions within a set of genes.

LLM Based Gene Set Generation Enables Steerable Overrepresentation Analysis

After establishing that our LLM-based approach is more effective than using a single prompt to identify multiple biological processes within a gene set, we applied this method to the differentially expressed genes (DEGs) from an experiment involving microglia differentiated from induced pluripotent stem cells (iMGs), treated with AL002 (N=4) compared to an isotype control (N=4) (see Methods). AL002 is an investigational, humanized, TREM2-selective, agonistic monoclonal antibody in Phase 2 trials for treatment of early Alzheimer's disease [23]. Activation of microglia is hypothesized to be an important mechanism of AL002. In iMGs, we found that 30 genes were differentially expressed at an FDR of 10% between iMG that received AL002 to the isotype control (Fig. 4A).

Given that it is difficult to define ground truth overrepresented gene sets from RNA-seq data, we qualitatively compared the overrepresentation results of using a fixed database derived from KEGG to that of results using our proposal-based approach (Fig. 4B). Using the KEGG database, we found several highly ranked pathways that were irrelevant to microglia. These include pathways rheumatoid arthritis and measles which provide little information on the impact of AL002 *in vitro*. In contrast, a larger number of highly ranked pathways related to immune and osteoclast function were detected by *llm2geneset*, our proposal-based approach. The enrichment in osteoclast function was driven by DEGs such as *ACP5* and *DCSTAMP*, which have also been associated with functions in the human immune system [24–26]. In this setting, the LLM was unaware of the experimental context of the input DEGs.

To address this scenario, we further extended the gene set proposal prompt to incorporate contextual information which steered the model toward gene set descriptions related immune and microglial function (see Methods, Fig. 4D). Here, the gene set proposal prompt included the context "in vitro microglia treated with a TREM2 agonist antibody." Using this approach we found the biological process "regulation of inflammatory responses" was significant at FDR 1%. This result is consistent with TREM2's function of the innate immune system in the brain [27]. Pathways and biological processes related to immune function were ranked highly overall and gene sets related to osteoclast function were not tested. Overall, this example illustrates that LLMs can be steered to propose and generate gene sets highly relevant to a specific RNA-seq experiment. These gene sets can be evaluated for overrepresentation in a set of DEGs to facilitate interpretation of RNA-seq data sets and subsequent hypothesis generation.

Discussion

Our work demonstrates that LLMs are powerful tools for gene set generation for biological processes and pathways. We introduce an approach for assessing the quality of these generations using human curated databases of gene sets. We ask if the gene sets generated by LLMs using the same natural language description are overrepresented in human curated data sets. These gene sets have numerous applications. They can be used directly to score expression data sets for activity of these sets of genes. This can be done at the sample as well as single cell level. They can also be directly used in gene set enrichment methods including overrepresentation analysis or rank based methods such as GSEA, CAMERA, or ROAST [3–7]. Gene sets can go out of date, and this has been demonstrated to impact the results and interpretation of RNA-seq studies [28]. Language models will continue to be trained on updated scientific literature and thus may offer an alternative or complementary approach to gene set curation.

Language models have been demonstrated to possess emerging planning and reasoning capabilities [29]. This reasoning capability can be leveraged by prompting the model to infer the biological processes represented by a set of DEGs. The prompts to do so can elicit both the model's reasoning and confidence. To improve on a single prompt, we use an approach where we allow the model to propose gene set descriptions based on both the input set of DEGs and contextual information. Independent of any previous context, we ask the model to generate gene sets and then test if these gene sets are overrepresented in the original set of DEGs. This approach provides a statistical measure of whether the biological process or pathway is enriched in the input set of genes, which cannot be obtained by a single prompt. Although we can strongly steer the model with contextual information, this approach limits the impact of this steering by checking the model's gene set generations against the DEG input. We also demonstrate that this approach effectively accounts for the presence of multiple biological processes within a single set of genes.

LLMs encode knowledge from a vast, complex, and continually evolving body of literature on gene functions and biological processes. Given the flexibility with which his knowledge can be queried, LLMs have significant potential to enhance computational biology and bioinformatics algorithms. By incorporating LLM generation as a subroutine, these algorithms can seamlessly integrate prior knowledge about genes and their roles. Given their ubiquity and ease of use, LLMs are poised to become essential tools for interpreting high-throughput biological data. Moreover, they are likely to play a pivotal role in driving hypothesis generation, advancing biological discovery, and informing the development of novel therapeutic strategies.

Methods

LLMs and Gene Sets Databases

We used the following versions of OpenAI large language models: GPT-4o mini (gpt-4o-mini-2024-07-18), GPT-3.5 (gpt-3.5-turbo-0125), and GPT-4o (gpt-4o-2024-08-06). GMT files containing gene sets were downloaded from https://maayanlab.cloud/Enrichr/#libraries. We used the Reactome (Reactome_2022), KEGG (KEGG_2021_Human), and WikiPathways (WikiPathway_2023_Human) gene set databases for evaluation. Each of the gene set descriptions were cleaned to remove any unique identifiers. To favor reproducible outputs from the LLMs, we used the seed parameter in the model API. We obtained the list of valid human HGCN symbols using the biomaRt R package by querying the `hsapiens_gene_ensembl` database for protein coding gene symbols [30].

Gene Set Generation

We used the following prompt to generate gene sets:

```
List all the known genes directly and indirectly involved in the following
biological process or cellular component """{descr}""". Use the following
JSON schema:
```json
{{
    "type": "array",
    "items": {{
        "type": "object",
        "properties": {{
            "gene": {{
                "type": "string",
            }}
        }},
        "required": ["gene"]
    }}
}}
```
The field `gene` is a gene involved in the following biological process or
cellular component: """{descr}""". Use the HUGO Gene Nomenclature Committee
(HGNC) gene abbreviations. Place the output in a JSON code block. Do not add
any comments in the JSON code block.
```

In the above prompt `{descr}` was replaced with a natural language description of a gene set. We used the following role prompt which we placed in the system message:

```
You are an expert in cellular and molecular biology.
```

The https://pypi.org/project/json-repair/ package was used to parse out the returned genes and repair any minor JSON formatting errors. If the JSON output was not repairable, we queried the model again with a different seed value. We used the input and output tokens returned by the model API, accumulating additional tokens if additional model queries

were needed. We removed any duplicate gene symbols returned by the LLM. The above prompt, query to an LLM, and parsing can be concisely written as function that takes as input a string description of a gene set and returns a string list with the corresponding gene symbols:

```
G:List[str] <- GetGenes(D:str)
```

Evaluating Gene Sets Generated by LLM

We evaluated the gene sets generated using the prompt above by examining the overrepresentation gene sets generated by LLM in the gene set curated by a human with the same description. We computed the overlap between the gene sets returned by LLM and the human curated gene sets and used the hypergeometric distribution to perform a one-tailed Fisher's exact test to compute a p-value corresponding to the probability whether the overlap, or higher overlap, was observed by chance. We used the hypergeom.sf in the scipy.stats package to compute numerically accurate p-values. We assumed possible set of N=19,846 human protein coding genes based on the statistics of GRCh38.p14 (https://useast.ensembl.org/Homo_sapiens/Info/Annotation). We corrected for multiple testing across gene sets using Bonferroni correction or false discovery rate as indicated. To evaluate model generations, we report the fraction of significantly overrepresented gene sets after multiple testing correction. Multiple testing corrections were performed across all gene sets within a database.

Additional Gene Set Generation Prompting Strategies

Prompts for confidence and reasoning closely followed our original prompt above. We modified the query portion of the original prompt to elicit model confidence as follows:

```
List all the known genes directly and indirectly involved in the following
biological process or cellular component """{descr}""" and your confidence
low, medium, or high in that they belong.
```

The formatting portion of the prompt included modifications to the output JSON schema to capture this additional confidence information. Only high confidence genes were used in our evaluation. Model reasoning was elicited in a similar manner with the query portion of the prompt modified as follows:

```
List all the known genes directly and indirectly involved in the following
biological process or cellular component """{descr}""" and a one sentence
reason why they belong.
```

Here, we also modified the JSON schema in the formatting portion of the prompt to output such that the single sentence corresponding to the model reason was captured. Ensembling was conducted by performing 5 generations of gene sets with different seed values for the model to obtain different genes for each generation. Genes that were

consistently observed in all 5 generations were returned as the genes for the given gene set description.

Proposing Gene Set Descriptions from a Set of Genes

We used the following prompt to propose a set of gene set descriptions from a list of genes.

```
List {n_pathways} biological pathways, biological processes, or cellular
components that contain the following genes """{genes}""" with high
confidence. Be as specific as possible. List non-overlapping pathways,
processes, or components. Do not include the gene names in the outputs. Use
the following JSON schema:
```json
{{
    "type": "array",
    "items": {{
        "type": "object",
        "properties": {{
            "p": {{
                "type": "string",
            }},
        }},
        "required": ["p"]
    }}
}}
```
Example output will look like the following:
```json
[{{"p":"BP or Pathway 1"}},
 {{"p":"BP or Pathway 2"}},
 {{"p":"BP or Pathway 3"}},
 {{"p":"BP or Pathway 4"}}
```
The element `p` designates a pathway, biological process or cellular
component. Place the output in a JSON code block. Do not add any comments in
the JSON code block.
```

In the above prompt, `{n_pathways}` was replaced with the number of desired biological pathways and process descriptions. `{genes}` was replaced by a comma separated list of genes for which we requested these biological processes and pathways. As with the gene set generation prompt, we used the https://pypi.org/project/json-repair/ package to parse out the returned genes and repair any minor JSON formatting errors. If the JSON output was not repairable, we queried the model again with a different seed value.

To steer the model toward experimentally relevant biological processes and pathways we additionally modified the above prompt to include contextual information when this contextual information was provided:

```
List {n_pathways} biological pathways, biological processes, or cellular
components that contain the following genes """{genes}""" with high
```

```
confidence. Also consider the following context as related to the genes:
"""{context}""" when selecting pathways, processes, and components.
```

Here `{context}` was replaced with a user provided string that provided additional context to steer the model generations (e.g. “in vitro microglia treated with a TREM2 agonist antibody”). The above can be concisely written as a function that takes as input a list of genes D, a requested number of processes and pathways N, and optionally a C context string:

```
P:List[str] <- GetPathwaysProcesses(D:List[str], N:int, C:str)
```

Discovery of Multiple Enriched Biological Processes in Gene Sets

llm2geneset proposes pathways and biological process descriptions based on the input set of genes and an experimental context. These pathway descriptions, alone, are used to generate gene sets which are then tested for overrepresentation in the input set of DEGs. The `GetGenes()` function call below does not have access to any previous context. The parameters of the algorithm are as follows:

- D = set of DEGs (or any set of genes)
- N = number of biological pathways and processes to propose
- C = (optional) contextual information regarding the experiment from which the DEGs were obtained
- B = number of background gene sets for overrepresentation analysis

Pseudo-code is provided below:

```
llm2geneset(D:List[str], N:int, C:str)
      R = []
      P = GetPathwaysProcesses(D)
      for pathway in P:
            G = GetGenes(pathway)
            p = hgsf(|intersect(D,P)|- 1, B - |D|, |D|, |G|)
            R = R.append((pathway,p))
      return padjust(R)
```

In the above pseudo-code, the function `padjust()` computes q-values to account for multiple testing. `hgsf()` is one minus the cumulative distribution function (1-CDF) of the hypergeometric distribution. The function call computes the p-value according to a one-tailed Fisher’s exact test. *llm2geneset* returns a list of N pathways sorted on their overrepresentation p-values.

Parsing GSAI Prompt LLM Outputs

We used the previously published GSAI prompt [9,10]. The GSAI prompt provides extensive instructions to an LLM to provide a gene set description, a confidence value, and an

analysis of how genes belong to the returned gene set description. We used regular expressions to parse out the gene set name, confidence, and the analysis. When we were unable to parse these outputs, we queried the LLM again with a different seed value to obtain outputs that could be parsed by regular expressions.

Evaluating Biological Process Discovery from Gene Sets

We compared the performance of GSAI and *llm2geneset* through two experiments. In the first experiment, we used gene sets from KEGG, Reactome, and Wikipathways to evaluate how closely the outputs of the two methods matched the original gene set descriptions. The input gene sets were shuffled before being submitted to either GSAI or *llm2geneset*. We assessed similarity using three metrics: (1) the fraction of single words (unigrams) and (2) word pairs (bigrams) in the original descriptions that appeared in the returned descriptions, and (3) cosine similarity between text embeddings. Embeddings were computed using OpenAI's text-embedding-3-large model, and the dot product of the embeddings was used for comparison. The average similarity score across all gene sets was used to evaluate the two methods. For *llm2geneset*, which generates multiple gene sets (N=5), we only considered gene sets significant at a Bonferroni-corrected p=0.002. If no significant sets were found, we returned "None Found." The maximum score for each metric across the significant gene set descriptions was reported.

In the second experiment, we focused on gene sets where the cosine similarity between GSAI and *llm2geneset* descriptions was >0.7 in the first experiment. We randomly selected 50 such pairs, combined their genes, and shuffled the new gene sets. For GSAI, we used the parsed name from the LLM output, and for *llm2geneset*, we generated N=5 gene sets, again keeping only those significant at p=0.002. We combined the descriptions of the significant gene sets into a comma-separated string or returned "None Found" if no significant sets were identified. We compared the GSAI description and the *llm2geneset* description to a reference string, which combined the original human-curated descriptions in a comma-separated format. Similarity was quantified by the fraction of unigrams and bigrams from the reference string present in the GSAI and *llm2geneset* outputs, as well as the cosine similarity between the strings. The average similarity across all 50 combined gene sets was used to compare the two methods.

Bulk RNA-seq from iMGs Treated with AL002

Induced pluripotent stem cells (iPSCs) were differentiated into mature microglia as previously described [31] . A total of 2.5 x 10^4 cells each well were seeded in a 96-well plate format, and after 7 days in culture, the cells were treated with AL002 or an isotype-IgG control at 1ug/ml and incubated for an additional 24h. Next, cells were harvested, and RNA was isolated using the RNeasy 96 QIAcube HT Kit (Qiagen, Hilden, Germany). The quality of RNA was assessed using the TapeStation System (Agilent Technologies, Santa Clara, CA). RNA samples were converted to cDNA libraries using the Universal Plus mRNA-Seq kit with NuQuant (Tecan Genomics, Redwood City, CA). RNA sequencing was

performed on an Illumina NovaSeq 6000 system using a NovaSeq V1.5 kit (SeqMatic, Fremont, CA), with 2x100 bp paired-end reads.

Sequencing quality was evaluated using FastQC (v0.12.1). Raw reads were trimmed and filtered using fastp (v0.23.4) to remove low quality bases and adapter sequences, discarding reads shorter than 25 bp[32]. Trimmed reads were aligned to the reference GRCh38 transcriptome with STAR (v2.7.11b)[33]. Sequencing and alignment quality control was summarized with MultiQC (v1.19)[34]. Differential expression analysis was conducted using DESeq2[35].

Code and Data Availability

All code, notebooks, and benchmarking data sets are available here:
https://github.com/Alector-BIO/llm2geneset.
All prompts used in this study are available for download here:
https://github.com/Alector-BIO/llm2geneset/tree/main/src/llm2geneset/prompts
A streamlit web application incorporating this tool is available here:
https://github.com/Alector-BIO/llm2geneset/tree/main/webapp.
An instance of the web application can also be accessed here:
https://llm2geneset.streamlit.app.

Conflict of Interest Statement

All authors are current employees of Alector Inc.

Author Contributions

ZK and JZ conceived of the idea. ZK, JZ, RYW, and AM contributed ideas and analyses. XW, RA, and JK contributed data. All authors read and approved the final manuscript.

Figures

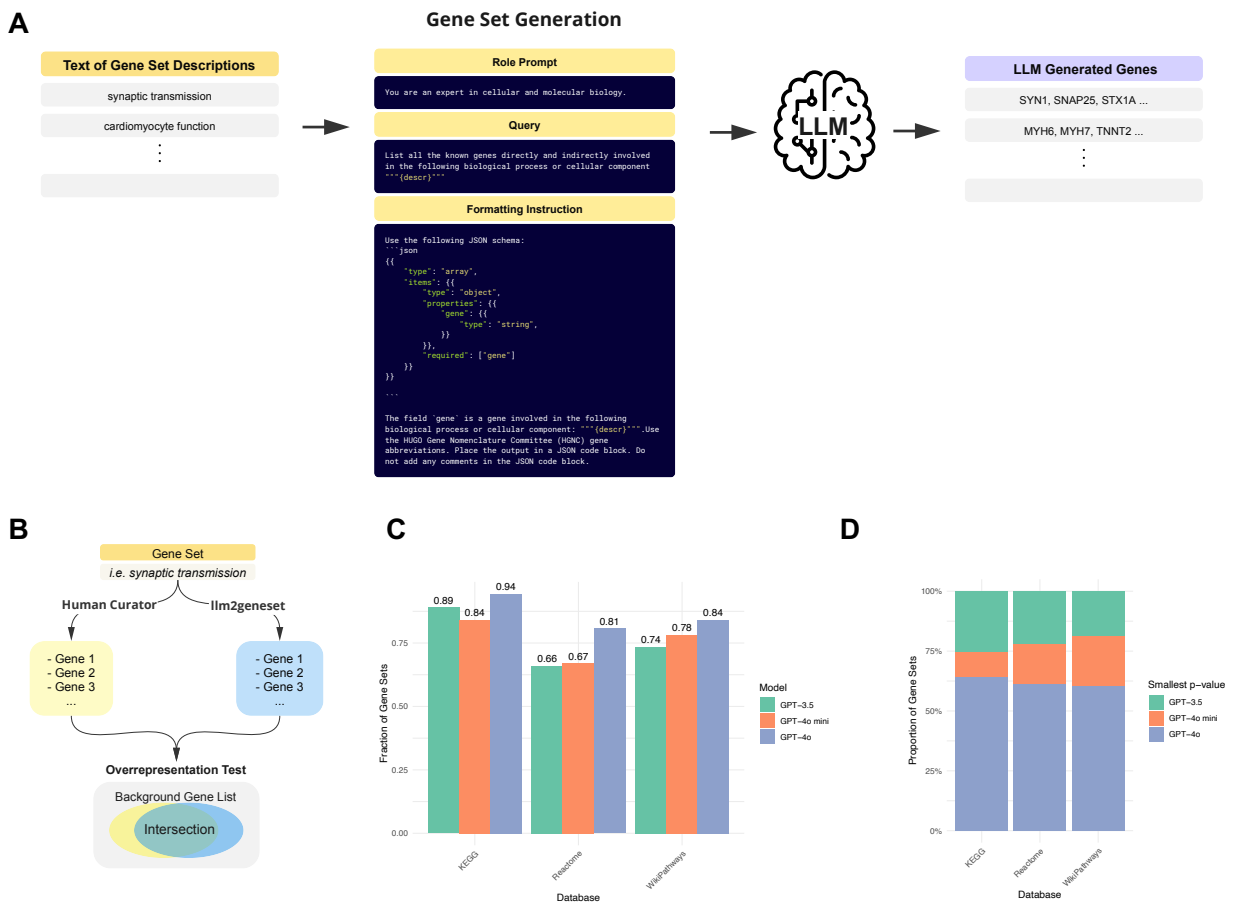

Figure 1. Evaluating gene sets generated by LLMs. (a) Diagram illustrating our prompt for generating gene sets based on a natural language description of the gene set. The prompt consists of several components including the request itself as well as formatting instructions. (b) Gene sets generated by LLM can be evaluated for significant overrepresentation in a human curated gene set. Significant overrepresentation indicates the LLM captures the relationship between the gene set description and corresponding genes within the gene set. (c) Fraction of gene sets that shows significant enrichment in the human curated gene sets at a Bonferroni adjusted p-value of 0.01 across databases and LLMs. (c) Proportion of gene sets generated by LLM where the overrepresentation p-value is smallest for a given model.

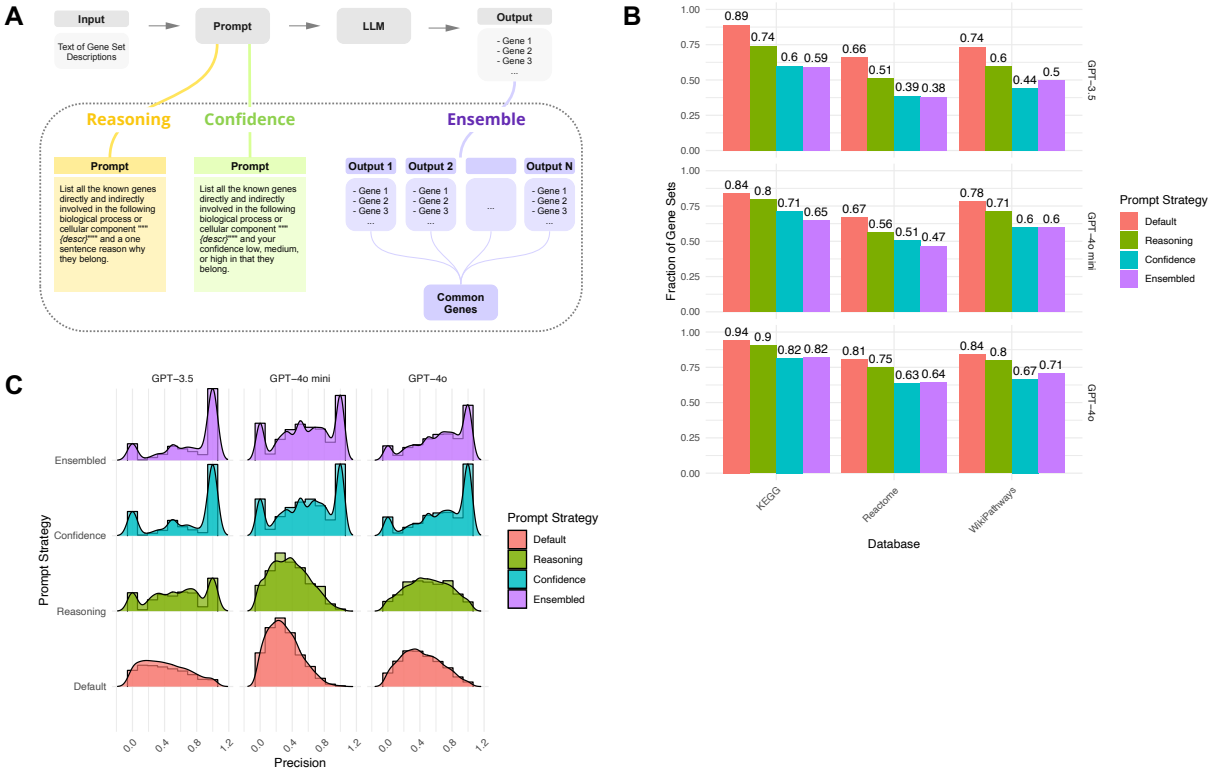

Figure 2. Assessment of Model Reasoning, Confidence, and Ensembling for Gene Set Generation. (a) Strategies to generate gene sets using an LLM prompt. (b) Fraction of genes where the LLM generated gene set is significantly overrepresented in the human curated gene set with the same natural language description. Bar plot color designates one of 3 prompting strategies illustrated in (a). (c) Precision of the gene sets returned or the fraction of LLM generated genes in a gene set that are in the human curated gene sets with the same description.

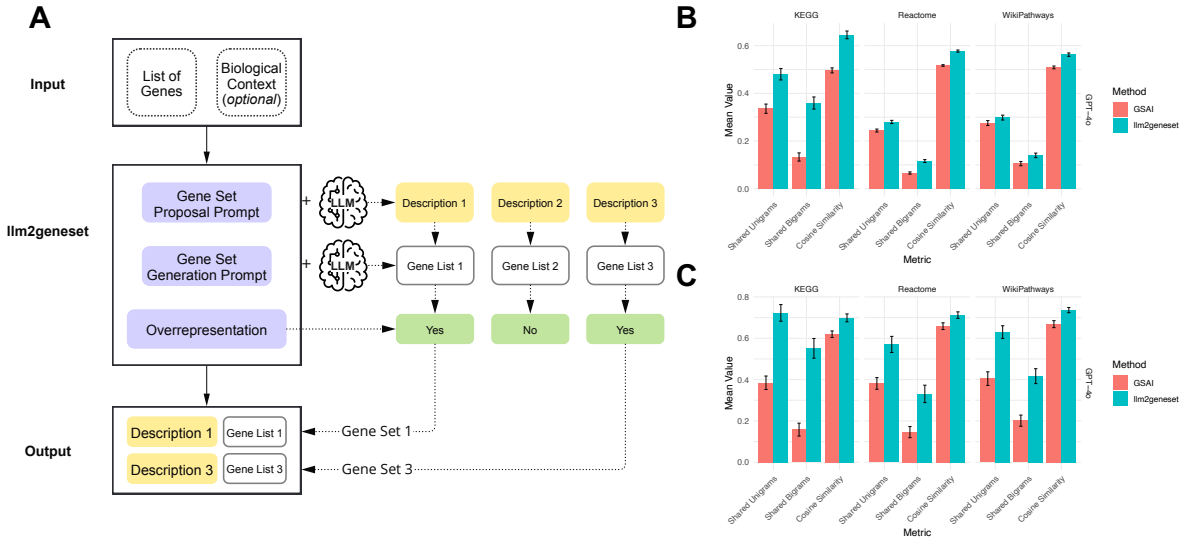

Figure 3. Gene Set Proposal Enables Discovery of Multiple Enriched Biological Functions. (a) *llm2geneset*, our gene set proposal-based strategy for determining the functions present in a set of genes. An LLM is used to propose gene set descriptions. These descriptions are then used to generate gene sets which are tested for overrepresentation with the input gene set. (b) Mean fraction of shared unigrams (words) and bigrams (word pairs) and mean cosine similarity between gene set descriptions returned by the GSAI prompt and *llm2geneset* across databases for the GPT-4o model. Error bars designate the standard error of the mean. (c) Mean fraction of shared unigrams (words) and bigrams (word pairs) and mean cosine similarity between gene set descriptions returned by GSAI and *llm2geneset* and the gene set descriptions originally assigned to mixed gene sets. Error bars designate the standard error of the mean.

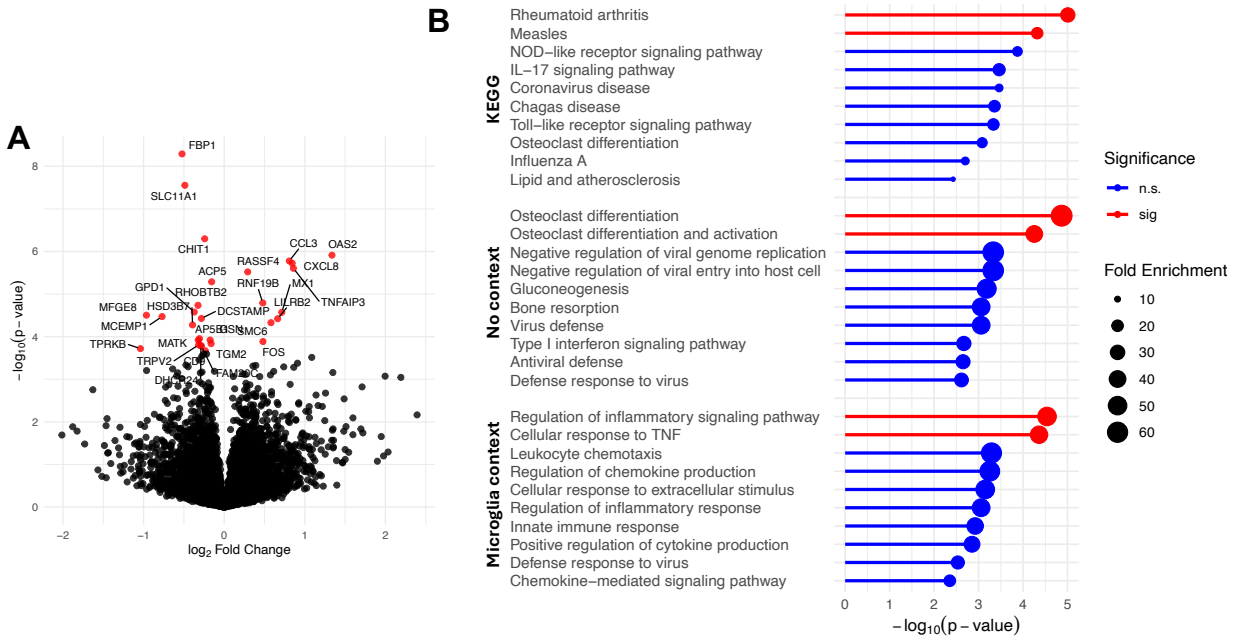

Figure 4. Comparison of overrepresentation analyses using static and LLM generated gene sets. (a) Volcano plot of DEGs comparing iPSC derived microglia treated with a TREM2 agonist antibody (AL002) to an IgG control antibody at 1ug/ml for 24 hours. Differentially expressed genes at FDR 10% are shown as red points. (b) Overrepresentation analysis results for biological process and pathways in: (top) the KEGG database; (middle) GPT-4o generated gene set database using no contextual information about the experiment; (bottom) GPT-4o generated gene set database using the contextual text "in vitro microglia treated with a TREM2 agonist antibody." The lollipop plot indicates the -$\log_{10}$(p-value) for overrepresentation of the gene sets in the set of DEGs in (a). Size of the circles in the plot designates fold enrichment of the gene set. Significant (sig) gene sets at an FDR of 1% are shown in red. non-significant (n.s.) gene sets are shown in blue.

References

1. Mubeen S, Tom Kodamullil A, Hofmann-Apitius M, Domingo-Fernández D. On the influence of several factors on pathway enrichment analysis. Briefings in Bioinformatics. 2022;23: bbac143. doi:10.1093/bib/bbac143

2. Maleki F, Ovens K, Hogan DJ, Kusalik AJ. Gene Set Analysis: Challenges, Opportunities, and Future Research. Front Genet. 2020;11. doi:10.3389/fgene.2020.00654

3. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. Proceedings of the National Academy of Sciences. 2005;102: 15545–15550. doi:10.1073/pnas.0506580102

4. Huang DW, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. Nat Protoc. 2009;4: 44–57. doi:10.1038/nprot.2008.211

5. Kuleshov MV, Jones MR, Rouillard AD, Fernandez NF, Duan Q, Wang Z, et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. Nucleic Acids Res. 2016;44: W90–W97. doi:10.1093/nar/gkw377

6. Wu D, Lim E, Vaillant F, Asselin-Labat M-L, Visvader JE, Smyth GK. ROAST: rotation gene set tests for complex microarray experiments. Bioinformatics. 2010;26: 2176–2182. doi:10.1093/bioinformatics/btq401

7. Wu D, Smyth GK. Camera: a competitive gene set test accounting for inter-gene correlation. Nucleic Acids Res. 2012;40: e133. doi:10.1093/nar/gks461

8. Hou W, Ji Z. Assessing GPT-4 for cell type annotation in single-cell RNA-seq analysis. bioRxiv. 2023; 2023.04.16.537094. doi:10.1101/2023.04.16.537094

9. Hu M, Alkhairy S, Lee I, Pillich RT, Bachelder R, Ideker T, et al. Evaluation of large language models for discovery of gene set function. ArXiv. 2023; arXiv:2309.04019v1.

10. Hu M, Alkhairy S, Lee I, Pillich RT, Fong D, Smith K, et al. Evaluation of large language models for discovery of gene set function. arXiv; 2024. doi:10.48550/arXiv.2309.04019

11. Jin Q, Yang Y, Chen Q, Lu Z. GeneGPT: augmenting large language models with domain tools for improved access to biomedical information. Bioinformatics. 2024;40: btae075. doi:10.1093/bioinformatics/btae075

12. Sahoo P, Singh AK, Saha S, Jain V, Mondal S, Chadha A. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications. arXiv; 2024. doi:10.48550/arXiv.2402.07927

13. Schulhoff S, Ilie M, Balepur N, Kahadze K, Liu A, Si C, et al. The Prompt Report: A Systematic Survey of Prompting Techniques. In: arXiv.org [Internet]. 6 Jun 2024 [cited 8 Sep 2024]. Available: https://arxiv.org/abs/2406.06608v3

14. Kanehisa M, Goto S. KEGG: kyoto encyclopedia of genes and genomes. Nucleic Acids Res. 2000;28: 27–30. doi:10.1093/nar/28.1.27

15. Joshi-Tope G, Gillespie M, Vastrik I, D'Eustachio P, Schmidt E, de Bono B, et al. Reactome: a knowledgebase of biological pathways. Nucleic Acids Res. 2005;33: D428-432. doi:10.1093/nar/gki072

16. M M, A A, A R, A W, Dn S, K H, et al. WikiPathways: connecting communities. Nucleic acids research. 2021;49. doi:10.1093/nar/gkaa1024

17. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, et al. Language Models are Few-Shot Learners. arXiv; 2020. doi:10.48550/arXiv.2005.14165

18. OpenAI, Achiam J, Adler S, Agarwal S, Ahmad L, Akkaya I, et al. GPT-4 Technical Report. arXiv; 2024. doi:10.48550/arXiv.2303.08774

19. Chen L, Zaharia M, Zou J. FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance. arXiv; 2023. doi:10.48550/arXiv.2305.05176

20. Pope R, Douglas S, Chowdhery A, Devlin J, Bradbury J, Levskaya A, et al. Efficiently Scaling Transformer Inference. arXiv; 2022. doi:10.48550/arXiv.2211.05102

21. Wei J, Wang X, Schuurmans D, Bosma M, Ichter B, Xia F, et al. Chain-of-thought prompting elicits reasoning in large language models. Proceedings of the 36th International Conference on Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates Inc.; 2024. pp. 24824–24837.

22. Cao H. Recent advances in text embedding: A Comprehensive Review of Top-Performing Methods on the MTEB Benchmark. arXiv; 2024. doi:10.48550/arXiv.2406.01607

23. Jackson S, Paul R, Joshi A, Liao Y, Morrison G, Castro A, et al. INVOKE-2 - A Phase 2 Randomized, Double-Blind, Placebo-Controlled Study to Evaluate the Efficacy and Safety of AL002 in Participants with Early Alzheimer's Disease (P17-3.005). Neurology. 2022;98: 2423. doi:10.1212/WNL.98.18_supplement.2423

24. Yagi M, Miyamoto T, Sawatani Y, Iwamoto K, Hosogane N, Fujita N, et al. DC-STAMP is essential for cell–cell fusion in osteoclasts and foreign body giant cells. J Exp Med. 2005;202: 345–351. doi:10.1084/jem.20050645

25. Briggs TA, Rice GI, Daly S, Urquhart J, Gornall H, Bader-Meunier B, et al. Tartrate-resistant acid phosphatase deficiency causes a bone dysplasia with autoimmunity and a type I interferon expression signature. Nat Genet. 2011;43: 127–131. doi:10.1038/ng.748

26. Garcia-Hernandez M de la L, Rangel-Moreno J, Garcia-Castaneda M, Kenney HM, Paine A, Thullen M, et al. Dendritic cell-specific transmembrane protein is required for synovitis and bone resorption in inflammatory arthritis. Front Immunol. 2022;13: 1026574. doi:10.3389/fimmu.2022.1026574

27. Ulland TK, Colonna M. TREM2 - a key player in microglial biology and Alzheimer disease. Nat Rev Neurol. 2018;14: 667–675. doi:10.1038/s41582-018-0072-1

28. Wadi L, Meyer M, Weiser J, Stein LD, Reimand J. Impact of outdated gene annotations on pathway enrichment analysis. Nat Methods. 2016;13: 705–706. doi:10.1038/nmeth.3963

29. Wei J, Tay Y, Bommasani R, Raffel C, Zoph B, Borgeaud S, et al. Emergent Abilities of Large Language Models. In: arXiv.org [Internet]. 15 Jun 2022 [cited 4 Oct 2024]. Available: https://arxiv.org/abs/2206.07682v2

30. Durinck S, Spellman PT, Birney E, Huber W. Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt. Nat Protoc. 2009;4: 1184–1191. doi:10.1038/nprot.2009.97

31. Haenseler W, Sansom SN, Buchrieser J, Newey SE, Moore CS, Nicholls FJ, et al. A Highly Efficient Human Pluripotent Stem Cell Microglia Model Displays a Neuronal-Co-culture-Specific Expression Profile and Inflammatory Response. Stem Cell Reports. 2017;8: 1727–1742. doi:10.1016/j.stemcr.2017.05.017

32. Chen S, Zhou Y, Chen Y, Gu J. fastp: an ultra-fast all-in-one FASTQ preprocessor. Bioinformatics. 2018;34: i884–i890. doi:10.1093/bioinformatics/bty560

33. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, et al. STAR: ultrafast universal RNA-seq aligner. Bioinformatics. 2013;29: 15–21. doi:10.1093/bioinformatics/bts635

34. Ewels P, Magnusson M, Lundin S, Käller M. MultiQC: summarize analysis results for multiple tools and samples in a single report. Bioinformatics. 2016;32: 3047–3048. doi:10.1093/bioinformatics/btw354

35. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 2014;15: 550. doi:10.1186/s13059-014-0550-8

Supplementary Figures

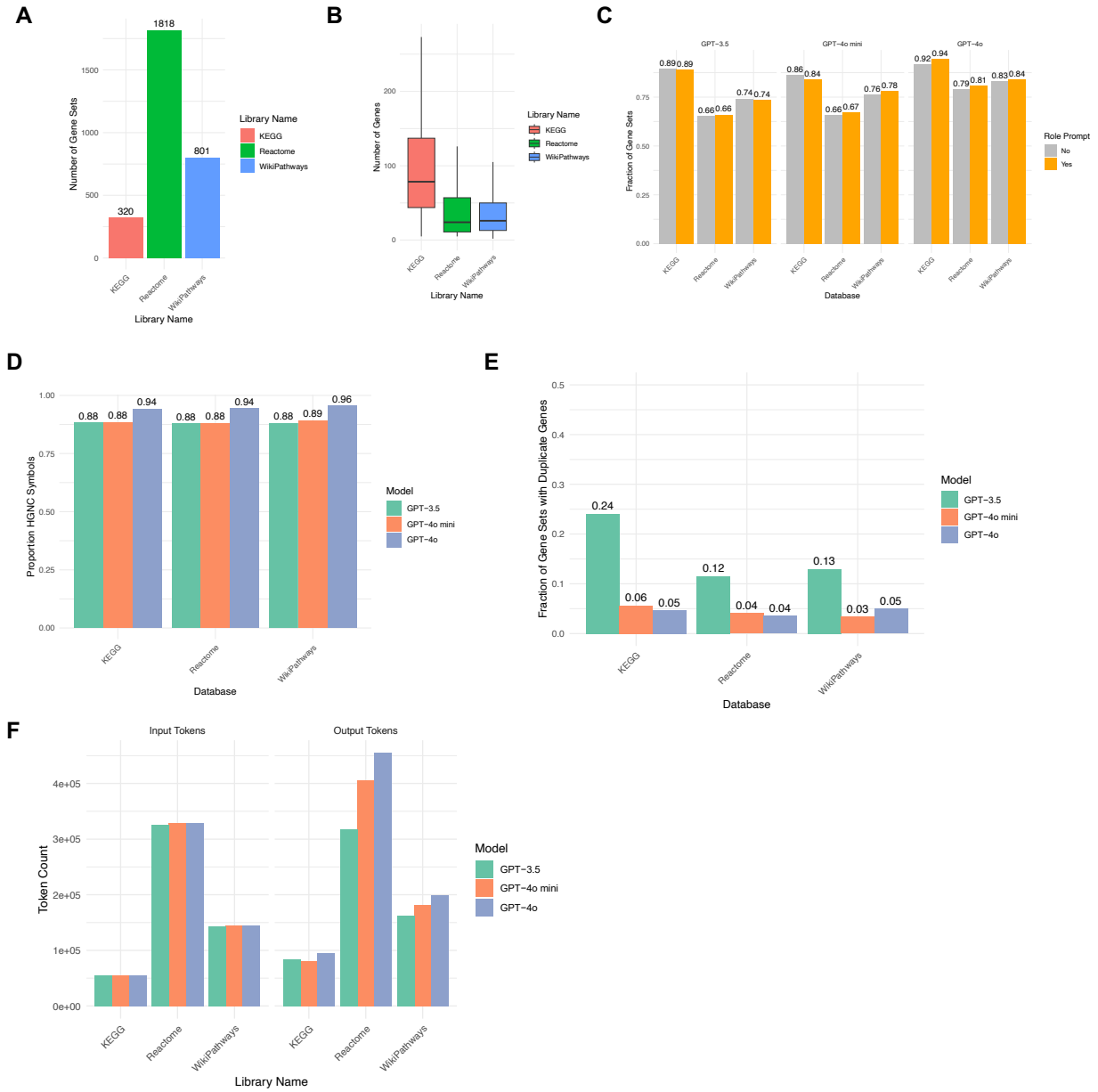

Figure S1. (a) Number of gene sets in the Reactome, Wikipathways and KEGG databases used for benchmarking. (b) Boxplot of the number of genes in each gene set across gene set databases. (c) Fraction of gene sets that show significant enrichment in human curated gene sets at a Bonferroni adjusted p-value of 0.01 across databases and LLMs. Plot compares the enrichment between gene sets generated using a role prompt and gene sets generated without a role prompt. (d) Fraction of gene symbols returned by LLMs across gene set databases that were known HGCN symbols in the Ensembl (GRch38.p14) genome annotation. (e) Fraction of gene sets that contain duplicate genes across LLMs for each of the benchmark gene sets. (f) Total number of input and output tokens used to generate gene sets across models and databases.

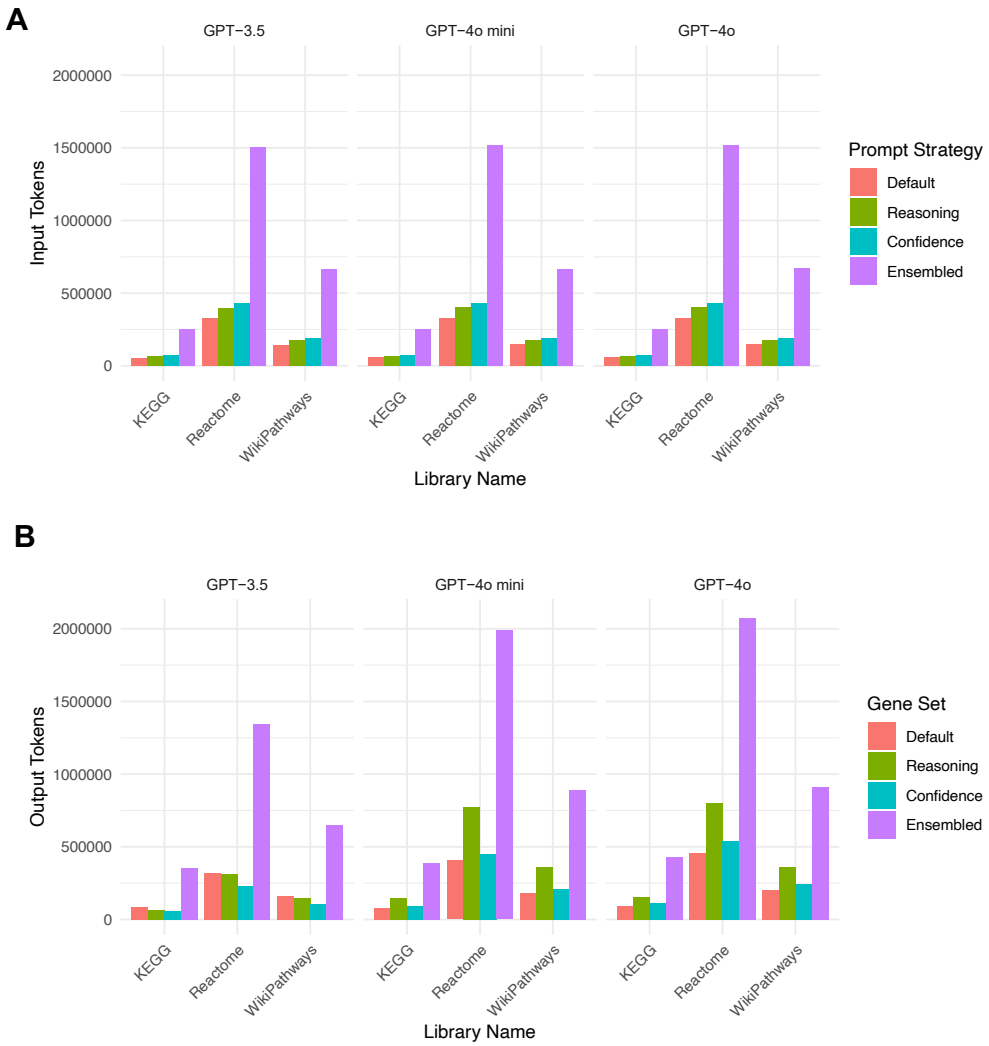

Figure S2. (a) Input tokens and (b) output tokens used by our prompting strategy in Figure 1 (Default prompt) as compared to a prompting strategy that incorporates model reasoning, self-reported confidence, and ensembling.

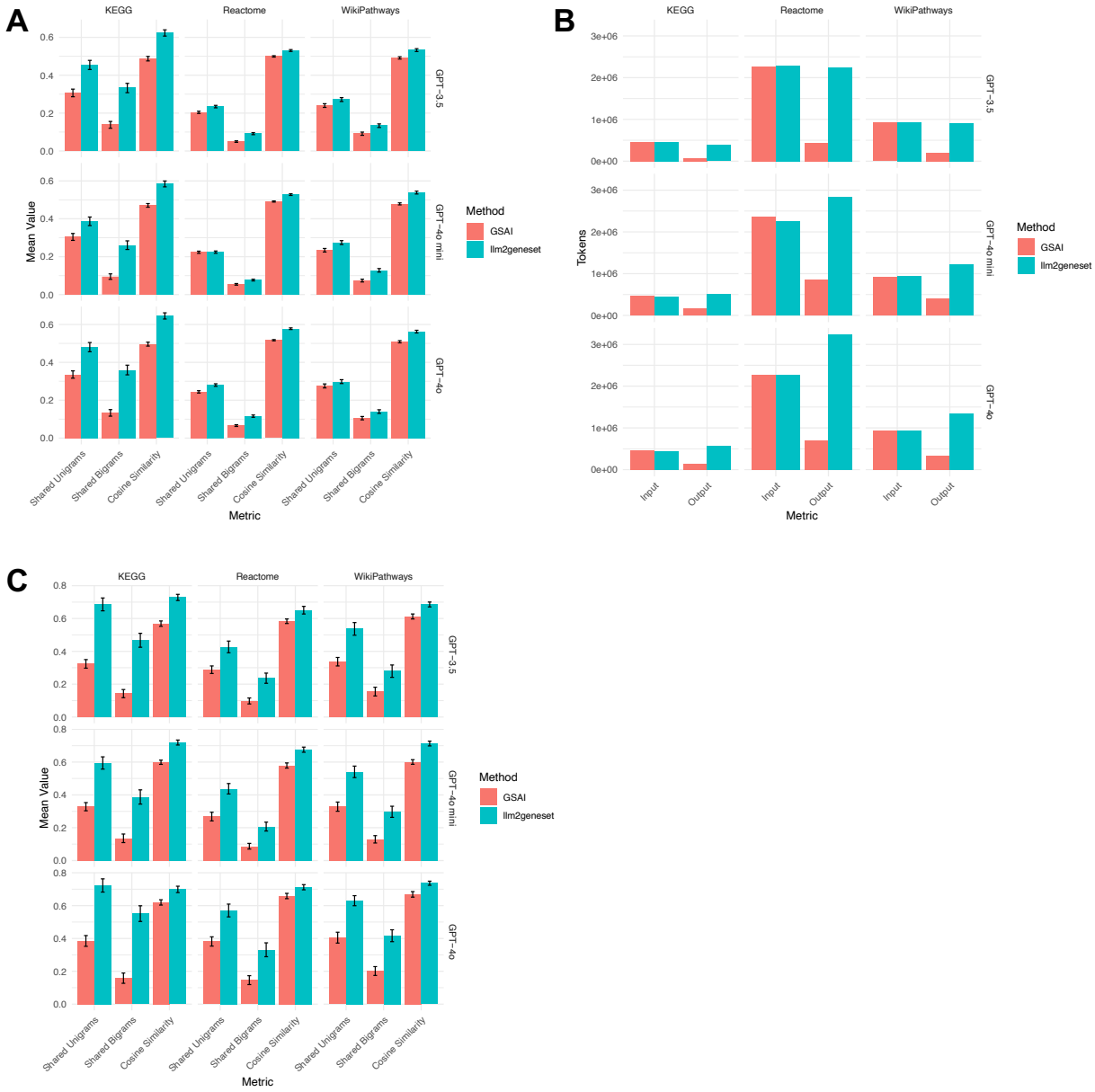

Figure S3. (a) Mean fraction of shared unigrams (words) and bigrams (word pairs) and mean cosine similarity between gene set descriptions returned by the GSAI prompt and llm2geneset across databases and LLMs. Error bars designate the standard error of the mean. (b) Input and output token usage of GSAI and llm2gene set across databases and models. (c) Mean fraction of shared unigrams (words) and bigrams (word pairs) and mean cosine similarity between gene set descriptions returned by GSAI and llm2geneset and the gene set descriptions originally assigned to mixed gene sets. Error bars designate the standard error of the mean.