

Summarization of BLAST Results Utilizing LLMs

Dr. Daniel Standage

Grace Frizzell

AI and machine learning in bioinformatics

- Genomic sequencing
- Protein and molecular structure modeling and predictions
- Data science
 - Understanding trends in large datasets and making predictions
- Summarization
 - Medical notes
 - Research texts

What is BLAST?

- Basic local alignment search tool
- Matching DNA/RNA sequences to each other



Standard Protein BLAST


[blastn](#) **[blastp](#)** [blastx](#) [tblastn](#) [tblastx](#)

BLASTP programs search protein databases using a protein query. [more...](#)

[Reset page](#)

[Bookmark](#)

Enter Query Sequence


Enter accession number(s), gi(s), or FASTA sequence(s)  [Clear](#)

Query subrange 


From


To

Or, upload file

[Choose File](#) No file chosen 

Job Title

Enter a descriptive title for your BLAST search 

☐ Align two or more sequences 

Choose Search Set

Databases

☒ Standard databases (nr etc.): **New** ☐ Experimental databases

 **Try experimental clustered nr database** 
For more info see [What is clustered nr?](#)

The problem

- BLAST results have large numbers of lines with free text descriptions
- We want summarizations that:
 - Eliminate redundancies
 - Concisely communicate meaning
 - Provide a consensus understanding of the descriptions
 - Accurately summarize
 - Summarize over differing sequence queries
- Can we actually do it?
- Can we do it here?

What are large language models?

- ChatGPT 4, Llama, Claude, CoPilot, etc.
- Sets of neural networks with many parameters, trained on vast amounts of data
- Subdivide words to produce meaningful responses
 - Tokens
- Conversational, summarical, programming, and other implementations exist.
- Models that you can download and run without an internet connection exist.
- Massive research focus


Solution characterization

- A quality solution:
 - Summarizes accurately
 - Matches the format of the descriptions
 - Can handle large prompt inputs → large enough to handle to dozens or hundreds of top blast hits
 - Processes quickly
 - Can work locally and in an air-gapped system.
- ChatGPT 4 is the standard?
 - Doesn't have the same limitations
- Niche applications like DNA Bert, BioBert, and others exist for diverse applications.

Prompt structure

WP_075294369.1 YopT-type cysteine protease domain-containing protein
WP_135026487.1 YopT-type cysteine protease domain-containing protein
WP_136121334.1 YopT-type cysteine protease domain-containing protein
WP_148417923.1 YopT-type cysteine protease domain-containing protein
WP_198974060.1 YopT-type cysteine protease domain-containing protein
WP_143813432.1 YopT-type cysteine protease domain-containing protein, partial
WP_076090542.1 YopT-type cysteine protease domain-containing protein
ARA70419.1 hypothetical protein BTV67_07780
WP_179121451.1 YopT-type cysteine protease domain-containing protein
WP_108574942.1 YopT-type cysteine protease domain-containing protein
WP_075291725.1 YopT-type cysteine protease domain-containing protein
Q06277.2 RecName: Full=Protein adenyllyltransferase and cysteine protease IbpA; Short=HMW IgBP; AltName: Full=p120; Includes: RecName: Full=Protein adenyllyltransferase IbpA; AltName: Full=AMPylator IbpA; Includes: RecName: Full=Cysteine protease IbpA; Contains: RecName: Full=Protein p76 IgBP; AltName: Full=76 kDa antigen; Flags: Precursor
WP_010906460.1 YopT-type cysteine protease domain-containing protein

WP_189293179.1 30S ribosome-binding factor RbfA
MBV9291373.1 30S ribosome-binding factor RbfA
WP_225016960.1 30S ribosome-binding factor RbfA
WP_197679966.1 30S ribosome-binding factor RbfA
WP_097871684.1 30S ribosome-binding factor RbfA
WP_104094537.1 30S ribosome-binding factor RbfA
WP_055796831.1 30S ribosome-binding factor RbfA
WP_055814013.1 30S ribosome-binding factor RbfA
WP_125499798.1 MULTISPECIES: 30S ribosome-binding factor RbfA
WP_158034433.1 30S ribosome-binding factor RbfA
PSJ26155.1 30S ribosome-binding factor RbfA
WP_093498772.1 30S ribosome-binding factor RbfA
WP_078074786.1 30S ribosome-binding factor RbfA
WP_058742112.1 MULTISPECIES: 30S ribosome-binding factor RbfA
WP_022868084.1 30S ribosome-binding factor RbfA
WP_235489993.1 30S ribosome-binding factor RbfA



These are BLAST results. The first column is an accession number, and the second is a free text description of the hit. In as few words as possible, provide a sequence of words that could demonstrate what these results generally say across all the entries.

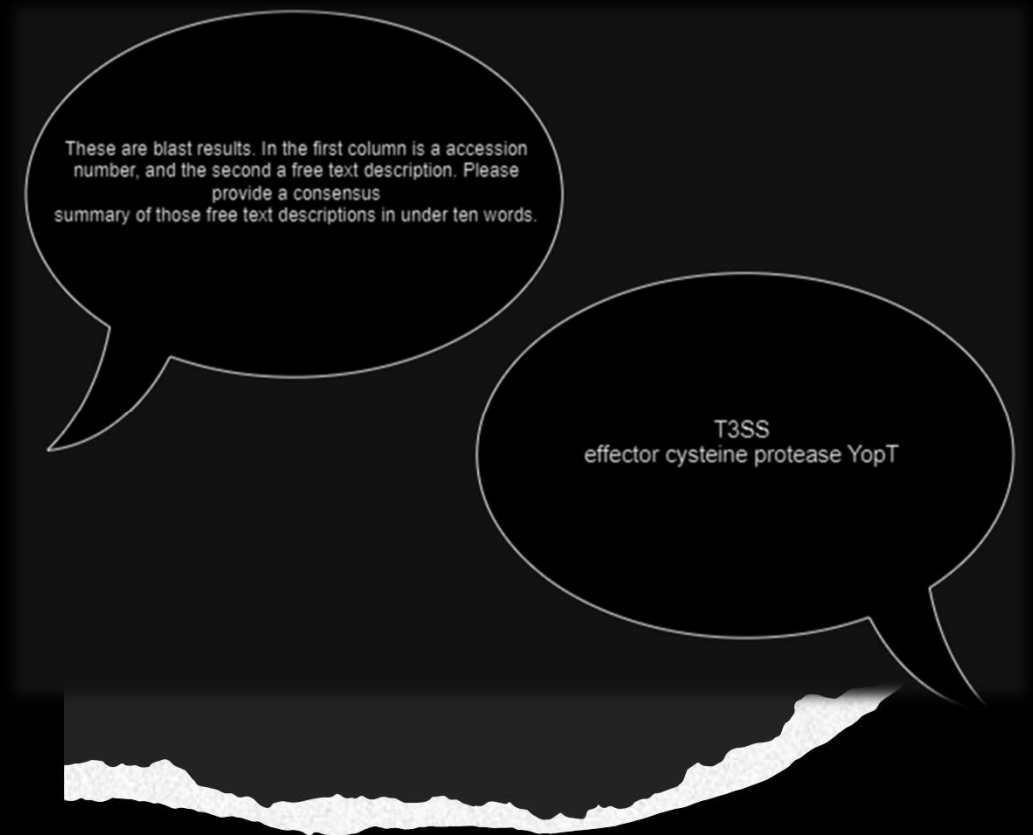
YopT-type cysteine protease domain proteins, hypothetical proteins, protease activity, protein adenyltransferase.

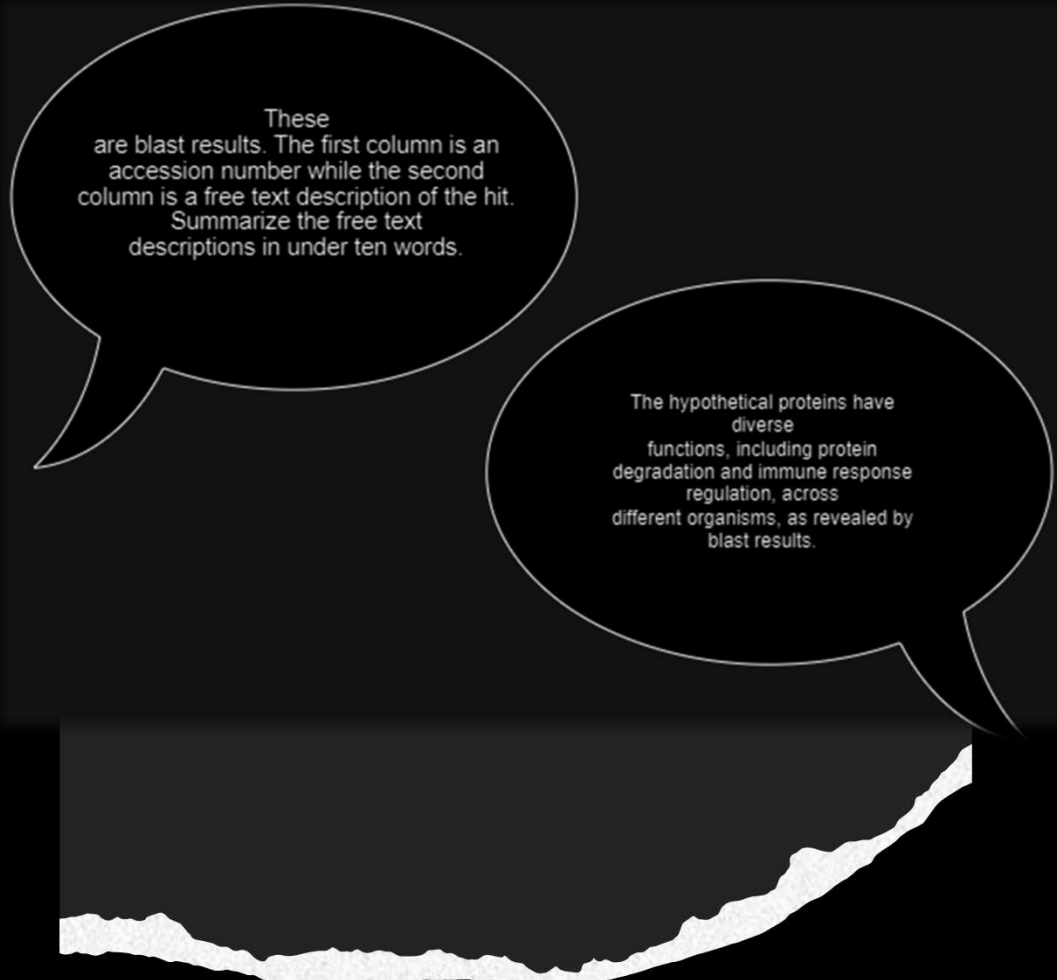
GPT 4

- Optimal prompts characterize the information and limit the response size
- Prompts can be as large as 2500 lines
- Response quality varies
- Golden standard for prompt size and context window
 - Not viable for airgapped system

Mistral Instruct

- 7 billion parameter model
- Significantly smaller context window
- Generally does best at concisely summarizing
- Ideal prompt characterizes the data and limits response scope





These
are blast results. The first column is an
accession number while the second
column is a free text description of the hit.
Summarize the free text
descriptions in under ten words.

The hypothetical proteins have
diverse
functions, including protein
degradation and immune response
regulation, across
different organisms, as revealed by
blast results.

Meta's Llama 2

- 7 billion parameter chat model
- Ideal prompt limits size and scope
- Struggles with concise summary

Common problems

- Summarization of information versus description of information
- Limited context windows and prompt sizes
- Need to issue other prompts to make response acceptable
 - GPT 4, Llama 2
- Integration with a solution
 - Diverse tools for diverse models
 - Lack of standardization

Iterative refinement



You

Summarize the content of the descriptions in under ten words.



ChatGPT

List of protein accession numbers with their associated types.



You

Don't describe what the information is like; summarize what the information is, in under ten words.



ChatGPT

Protein accession numbers for human, mouse, and rat sequences.



You

Summarize the content of the free text descriptions in under ten words.

FileEditSelectionViewGoRun...←→code

PROBLEMSOUTPUTDEBUG CONSOLETERMINALPORTS

Python + -

COMMENTS

Using fallback chat format: None

llama_print_timings: load time = 32472.84 ms

llama_print_timings: sample time = 2.50 ms / 19 runs (0.13 ms per token, 7606.08 tokens per second)

llama_print_timings: prompt eval time = 32472.49 ms / 211 tokens (153.90 ms per token, 6.50 tokens per second)

llama_print_timings: eval time = 4052.57 ms / 18 runs (225.14 ms per token, 4.44 tokens per second)

llama_print_timings: total time = 36585.19 ms / 229 tokens

A: The consensus summary is: "Protein structures with unknown function".

PS C:\Users\17176\OneDrive\Documents\Mount2023-2024\Internship\code>

There are no comments in this workspace yet.

main* 0 0 0 0Ln 47, Col 378Spaces: 4UTF-8CRLFPython 3.11.6 64-bit

Conclusions and going forward

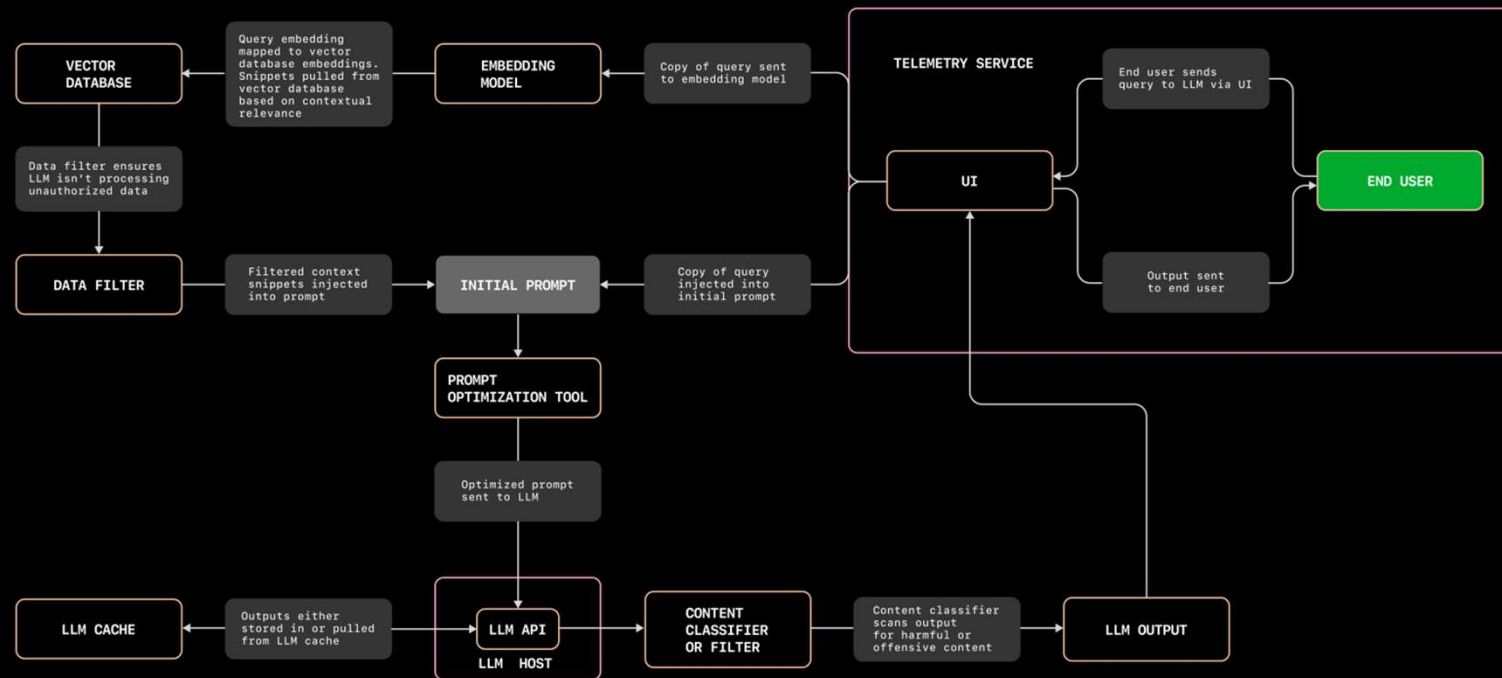
- Embedding tools and other implementations of LLM are diverse in methodology and level of support.
- LLMs constitute a tool of potential value in multiple fields and contexts.
- Exploring how we might use this tool promotes higher potential productivity across fields.
- Summarizing across sequences
- Integration with existing tools
- Construction of dedicated tools and applications

Questions?

AI and Machine Learning

- BioBert
- AlphaFold
- ESM2 and ESMFold
 - Meta
- MegaMolBart
- Splice AI

LLM Application Architectures



THIS DIAGRAM REPRESENTS THE ARCHITECTURE OF TODAY'S LLM APPLICATION. THE DIFFERENT COMPONENTS CAN BE ROUGHLY GROUPED INTO THREE CATEGORIES: USER INPUT, INPUT ENRICHMENT TOOLS AND PROMPT CONSTRUCTION, AND EFFICIENT AND RESPONSIBLE AI TOOLING.

LLM Embedding

- Embedding tools and other implementations of LLM are diverse in methodology and level of support.
- LLMs are just one implementation in a wider AI sphere.
- Often, one needs to customize the LLM.
 - Prompt-engineering
 - The “asks question” guy – John Stewart
 - Fine-tuning
 - Tailoring parameters to return results of a known truth value
 - Reward models
 - Response acceptance