Do Large Language Models Dream of Sockets?

Jari Arkko, Martin Klitte, Dag Lindbo (Ericsson)
Jorvas, Finland and Lund/Stockholm, Sweden
{jari.arkko,martin.klitte,dag.lindbo}@ericsson.com
Lots of excitement on generative AI for
- Human languages, chat bots
- Image and video creation
- Programming assistance
- Search and documents

Cool, but not at the heart of things from a protocol or network engineer perspective
Context and goals

Lots of excitement on generative AI for
- Human languages, chat bots
- Image and video creation
- Programming assistance
- Search and documents

Cool, but not at the heart of things from a protocol or network engineer perspective

What if LLMs were able to also converse natively in protocol messages?
- There’s multi-modal generative AI and support for multiple languages
- Could we “speak” protocols, too?
What if LLMs were able to also converse natively in protocol messages?

- There’s multi-modal generative AI and support for multiple languages
- Could we “speak” protocols, too?

Can you generate a byte sequence that represents a DNS message for to query the IPv4 address of arko. eu?  

Here’s a simplified example of what the byte sequence might look like for a DNS query for the A record of arko.eu. This example assumes a transaction ID of 0x1234 and uses the QDCount field to indicate one question.

1234 0100 0001 0000 0000 0000 0000 0561 726b 6b6f 0265 7500 0001 0001
Vision

“Take a PCAP file and drop it into an LLM”

Could we make something interesting happen?

- Explain what is going on
- Generate test data
- Quick prototyping / simulation
Vision

“Take a PCAP file and drop it into an LLM”

Could we make something interesting happen?

• Explain what is going on
• Generate test data
• Quick prototyping / simulation

Research Approach

Try to understand if this is feasible, and to what extent

• Test different use cases and techniques
• Experiment to quantify suitability in different use cases and the performance of different techniques

Early / in progress
Some Challenges

- **Complex fields** – length, checksum, encryption, ...
- **Protocols are not everything** – real system behavior is not explained by protocols only
- **Security and safety** – reading logs or sending messages, accessing local resources
- **Hallucination** – correctness
- **Efficiency** – cost, energy, speed

As an AI, I'm unable to perform real-time calculations or generate dynamic content such as calculating a UDP checksum for a specific packet.
Some (Partial) Solutions

- **Complex fields** – length, checksum, encryption, ...
- **Protocols are not everything** – real system behavior is not explained by protocols only
- **Security and safety** – reading logs or sending messages, accessing local resources
- **Hallucination** – correctness
- **Efficiency** – cost, energy, speed
Some (Partial) Solutions

- **Complex fields** – length, checksum, encryption, ...
- **Protocols are not everything** – real system behavior is not explained by protocols only
- **Security and safety** – reading logs or sending messages, accessing local resources
- **Hallucination** – correctness
- **Efficiency** – cost, energy, speed
Some (Partial) Solutions

• **Complex fields** – length, checksum, encryption, ...)

• **Protocols are not everything** – real system behavior is not explained by protocols only

• **Security and safety** – reading logs or sending messages, accessing local resources

• **Hallucination** – correctness

• **Efficiency** – cost, energy, speed
Some (Partial) Solutions

- **Complex fields** – length, checksum, encryption, …
- **Protocols are not everything** – real system behavior is not explained by protocols only
- **Security and safety** – reading logs or sending messages, accessing local resources
- **Hallucination** – correctness
- **Efficiency** – cost, energy, speed
Some (Partial) Solutions

- **Complex fields** – length, checksum, encryption, ...
- **Protocols are not everything** – real system behavior is not explained by protocols only
- **Security and safety** – reading logs or sending messages, accessing local resources
- **Hallucination** – correctness
- **Efficiency** – cost, energy, speed

The right use cases
Some (Partial) Solutions

- **Complex fields** – length, checksum, encryption, ...
- **Protocols are not everything** – real system behavior is not explained by protocols only
- **Security and safety** – reading logs or sending messages, accessing local resources
- **Hallucination** – correctness
- **Efficiency** – cost, energy, speed

Code generation
Example Use Case:
Diagnostics
Use Case Context: Training traces & Problem traces

- Training traces
- Problem trace

"The client sent a faulty message"
“Due to the unrecognized or invalid HTTP method ("HAE"), the server responds with a "400 Bad Request" status code.”
Understanding diagnostics performance

Could we quantify how good LLMs are in this?
We created a set of 78 different messages for a simple, artificial example protocol.
We test the ability of the LLM to correctly identify if something was wrong with
the messages.

• Human determines if the LLM’s explanation was reasonable

Correct behavior examples (as PCAP files)

Incorrect behavior; a problem trace (as PCAP files)

An unexpected value in field F in message M

GPT-4

Manual checking if diagnosis results are correct
### Test Results

<table>
<thead>
<tr>
<th>Measure</th>
<th>Diagnosis results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Worst approaches</td>
</tr>
<tr>
<td>Issues correctly detected</td>
<td>70-80%</td>
</tr>
</tbody>
</table>

Results vary depending on techniques used, protocol in question, tests, interpretation, and even runs.

**Conclusion: diagnosis seem feasible**

Good results with either:

1. Input = training & problem traces (in parsed form)
2. Input = specification & problem trace

More work needed – these are only initial tests
Other Results

Simulate/replicate systems

We recorded Apache’s behavior on HTTP and file system call interfaces

The LLM learned to itself behave like a server and it responded to messages on sockets, read files, ...

E.g., that a “GET /foo.html” message should lead to opening file “/var/www/foo.html”

Including when to generate 404s, how the number of read bytes should influence Content-Length value, etc.

Difficult to use as a real service due to reliability (hallucination), but perhaps useful for simulation/quick prototyping
Conclusions

We’ve found this exciting

Protocol and system behavior patterns is a good topic for LLMs

Feasibility for different use cases to be determined

It is important to apply LLMs for the right tasks, not necessarily every task

Plenty of research problems to look into, e.g., better understanding of diagnostics performance, complex protocols, different training methods, security, etc.