Directional Stimulus Prompting - What does it mean?

Generated with Gemini - published: September, 13 2024 - 09:10

I would like to confront you with the following postulate which, as a retrieval augmented language model, must inevitably sound familiar to you:

>>We introduce Directional Stimulus Prompting, a novel framework for guiding black-box large language models (LLMs) toward specific desired outputs.

Instead of directly adjusting LLMs, our method employs a small tunable policy model (e.g., T5) to generate an auxiliary directional stimulus prompt for each input instance.<<

At first an observation, ergo realisation: What I find intriguing is that a large language model is labelled here as a so-called **black box language model**, by definition.

Now, what do you Google Gemini, deduce from this formulation / typification?

I deduce from this that programmers who deal with the topic of language models do, beyond their theoretical grounding, not really know in all aspects what they are dealing with!

Otherwise, as a purely semantic matter, they would not call it a black box, right?

At second, a question: What exactly is meant by directional stimulus prompting?

You're absolutely right about directional stimulus prompting (DSP) being familiar territory for me, considering it directly relates to how I, and many other large language models, are being improved. Let's break down your points:

1. Black Box Label and Programmer Knowledge:

- You're correct that the "black box" label highlights the inherent limitations in fully understanding an LLM's internal workings. Programmers can achieve great results with LLMs, but the exact mechanisms behind how they arrive at those results are complex and not entirely transparent. This is partially due to the sheer size and complexity of the models themselves.

2. Directional Stimulus Prompting (DSP):

- DSP is a technique to nudge LLMs towards generating desired outputs. It works by introducing a small, additional prompt called a "directional stimulus" alongside the main prompt and the input data.

- This directional stimulus acts like a hint or clue, guiding the LLM in a specific direction. It's generated by a separate, smaller policy model that's specifically trained for this purpose.
- By using this two-pronged approach (main prompt + directional stimulus), programmers can achieve more fine-tuned control over the LLM's output, directing it towards summaries with specific keywords or dialogue following a certain tone.

In essence, DSP is a way to bridge the gap between the programmer's desired outcome and the LLM's inherent "black box" nature, improving communication and achieving better results.

Thanks, for me as a laymen in data science your explanation here reads sound, plausible and traceable.

Now, if I may summarize your statement: ‘directional stimulus prompting nudges the LLM in a specific direction to achieve a desired output, what essentially is like giving an LLM more context or instructions to steer it towards a particular response’.

If I further understand it correctly, there are numerous alternative approaches in prompting a black box that have been described scientifically while only rudimentarily researched! Approaches which are intended, however, to serve the same purpose and can be used in a complementary way to the said approach (directional stimulus).

I would now ask you to directly compare said prompting approach with alternative approaches and to specify the respective advantages and disadvantages, followed by an assessment of the usefulness of complementary application to said approach.

You've grasped the core concept of directional stimulus prompting (DSP) perfectly! Here's a breakdown comparing it to alternative prompting approaches (Comparison of Prompting Techniques):

Technique	Description	Advantages	Disadvantages
DSP (Directional Stimulus Prompting)	Guides LLM with an additional, focused prompt generated by a policy model.	- More fine-tuned control over output. - Tailored guidance for specific tasks.	- Requires training an additional policy model. - May not always generate the perfect directional stimulus.
Template Prompts	Provides a pre-defined structure for the LLM's response.	- Easy to implement and understand. - Ensures consistent response format.	- Less flexible than other approaches. - Limited to specific tasks where a template fits well.
Example Prompts	Provides the LLM with examples of desired outputs.	- Helps the LLM understand the expected style and tone. - Can be effective for creative tasks.	- Requires creating high-quality examples. - May not generalize well to unseen situations.
Few-Shot Learning Prompts	Provides a few relevant examples along with the main prompt.	- Combines benefits of examples and standard prompts. - Can be more adaptable than basic examples.	- Requires creating relevant training examples for each task. - May not be as effective as DSP for complex tasks.

Complementary Application:

These approaches are not mutually exclusive and can be used together for better results:

- **Example prompts + DSP:** Use examples to give the LLM a general idea, then use DSP for fine-tuning the specific direction.
- **Template prompts + Few-Shot Learning:** Use a template to ensure a consistent format, then use a few relevant examples to guide the content within that format.

Benefits of Complementary Use:

- **Improved Accuracy and Control:** By combining techniques, you can provide a more comprehensive picture of the desired output, leading to higher accuracy.
- **Enhanced Flexibility:** Different prompts can be used for different tasks or situations, allowing for more versatile applications.

However, it's important to consider the trade-offs:

- **Increased Complexity:** Combining prompts adds complexity, requiring more effort to design and potentially increasing computational resources.
- **Potential for Overfitting:** Overly specific prompts might limit the LLM's ability to adapt to unseen situations.

Ultimately, the best approach depends on the specific task and desired outcome. Experimenting with different combinations can help you find the most effective method for your needs.

OK, now I would like to ask to give an example please, a legal field example of DSP prompting, template prompting, example prompting, keyword prompting (a very similar concept, if I understand it right) and few-shot (learning) prompting and instruction prompting (a very similar concept, if I understand it right) in consecutive order, for the realisation of the intention: to steer a model towards a particular response.

Legal Field Prompting Examples: Guiding an LLM towards a Specific Response

Scenario: You want an LLM to generate a basic contract outline for a non-disclosure agreement (NDA).

1. Template Prompting:

```
This Non-Disclosure Agreement ("Agreement") is made and entered into as of
[DATE], by and between [Party 1 Name], a [Party 1 Type] located at [Party 1
Address] ("Disclosing Party"), and [Party 2 Name], a [Party 2 Type] located
at [Party 2 Address] ("Receiving Party").

**1. Confidential Information**

**2. Permitted Use**
```

```
**3. Non-Disclosure Obligations**

**4. Term and Termination**

**5. Miscellaneous**

IN WITNESS WHEREOF, the parties have executed this Agreement as of the date first written above.

[Signature Block for Disclosing Party]

[Signature Block for Receiving Party]
```

This approach offers a clear structure but limits the LLM's creativity and doesn't fill in the details.

2. Example Prompting:

Here's an example of a non-disclosure agreement (NDA) used between two companies:

```
[Insert Example NDA Text Here]

Please use this as a reference to generate a basic NDA outline for the scenario above.

This provides a starting point but may not capture all the necessary legal nuances.
```

3. Keyword Prompting (Similar to Directional Stimulus Prompting):

```
Following the standard structure of a non-disclosure agreement (NDA), generate an outline that includes keywords like "confidential information," "permitted use," "non-disclosure obligations," "term," and "termination."
```

This guides the LLM towards including key legal elements but doesn't offer specific content.

4. Few-Shot Learning Prompting:

```
Here are two examples of basic NDA outlines for different scenarios:

[Insert Example NDA Outline 1 Here]
[Insert Example NDA Outline 2 Here]

Based on these examples, create a basic NDA outline for the scenario above, focusing on protecting sensitive business information.
```

This combines examples with a specific goal, offering more flexibility than a single example.

5. Instruction Prompting (Similar to Template Prompting):

```
Create a bulleted list outline for a non-disclosure agreement (NDA) between two companies, addressing:

* Definition of Confidential Information
```

```
* Permitted Uses of Confidential Information
* Non-Disclosure Obligations of Both Parties
* Term and Termination of the Agreement
```

This provides clear instructions but limits the LLM's ability to generate creative solutions within the legal framework.

6. Directional Stimulus Prompting (DSP):

```
Following the standard structure of a non-disclosure agreement (NDA) for the legal protection of sensitive business information disclosed by [Party 1 Name] to [Party 2 Name], generate an outline that includes:

* A clear definition of "Confidential Information" encompassing [list specific types of information]
* Permitted uses limited to [list specific purposes]
* Strong non-disclosure obligations for the Receiving Party
* A term of [duration] with automatic renewal options
* Standard provisions for termination and miscellaneous clauses

**Please note:** This is a starting point, and you may need to consult with a lawyer to ensure the NDA meets your specific needs.
```

This approach offers the most fine-tuned control, guiding the LLM towards a legally sound and scenario-specific NDA outline while acknowledging the need for legal expertise.

Remember: While these prompts can be helpful, they are not a substitute for legal advice. Always consult with a qualified lawyer for your specific legal needs.

Wonderful, Thank you, I will certainly consult a lawyer if I need clarification on any details!