Greetings and welcome to the AI-Operations team at Softage Information Technology Limited! We are thrilled to have you join us. As a valuable member of our AI-Operations team, your primary focus will be on Prompt Generation.

This document is designed to fine-tune your writing skills to align with our objectives and aspirations. Upon completion, you will be paired with a mentor who will further guide your journey with us. Now, let's get to the basics of prompting.

What is a prompt?

- A prompt is the initial input and instructions that guide a Large Language Model (LLM) towards generating a desired output. The effectiveness of an LLM depends heavily on the quality of its prompt. A well-crafted prompt can steer the model toward generating accurate, relevant, and creative outputs. At the same time, a poorly worded one can lead to misunderstandings and unsatisfactory results.
- In simpler terms, a prompt is the "question" you input into the LLM.

Use case for prompts:

- The prompts we engineer will primarily be used to train LLMs to rival industry-leading models like ChatGPT and Bard. Our goal is to figure out the places(domains) where the LLM is lacking polish, and engineer prompts to correct the LLM in those domains. Take a proactive approach in this, try different types of prompts, see if the LLM is lacking in any aspect, then make prompts to overcome those shortcomings.
- A good prompt will guide these models toward producing the desired output through carefully constructed instructions and contextual information.
- One key difference between the general use prompts and the prompts engineered by us is that our prompts will be used to train LLMs, while the other ones are for your personal use to get the best possible output from the LLM. We work for the LLM, while the LLM works for its users.
- Constraints are very important in making prompts to train LLMs. Constraints refer to any limitations or boundaries placed on the output or behavior of LLMs. These limitations can be imposed in various ways and serve different purposes, but all aim to guide the LLM towards more desirable or ethical outcomes.

- In simpler terms constraints are the “rules” we set up for the LLM so that it doesnt get lost in the vastly open world of answers. A clear set of rules will guide it towards the end goal, think of it like directions in a maze, which will guide you towards escaping it.

Let us look at the different elements in a prompt:

A good prompt needs to have, at the very least, these crucial elements:

1. **The task**:

- What specific “task/tasks” are you expecting the LLM to do for you? Have a clear idea about this, as this will be one of the two most crucial aspects in deciding if your prompt works.
- This can be anything from writing a poem, asking for a specific bandaging technique, debugging a problematic code, to asking for a PowerPoint presentation on a niche subject.
    - **prompt:** I broke my two front teeth and now want to know about 4 dentist-approved general options to fix it, costing not more than 300 dollars in Beverly Hills.
    - Here, the **task** is “find general options to fix the two front teeth”

2. **The necessary context**: A good prompt will have only the “relevant” information to help the LLM with response generation. Avoid “oversharing,” and only give the necessary context relevant to the task you are asking.

- Depending on the task, this can constitute some necessary background information, a goal you aim to achieve, a specific code you aim to emulate, or more.
    - **prompt:** “as the developer of the new policy in your company and want to create a policy for BYOD. Write 10 points that can be mentioned as do’s and don’t for the devices of the employee in the company. the points is specifcally about data security of company.”
    - **Context** here is “developer wanting to create a policy for BYOD”

3. **The constraints**:

- Having a clear set of constraints helps steer the LLM to give you a specific but tough answer you are looking for. Without constraints, the LLM can give you vague and generic answers. Adding clear constraints will help the LLM focus on the output you expect from it.
- This can be stuff like using a specific rhyming scheme for content, the amount of supplies and time you have to get a project done, asking to avoid a specific language when coding, or more.

- ○ **prompt:** “as tom hansen from 500 days of summer, Write a Cheerful bridge for the song after the second paragraph. Add a jazz tune in the background to make it sound more refreshing. add the smiths influences”
- ○ the **constraints** in this prompt are: while writing a bridge for the song, (i) be tom hansen from 500 days of summer, (ii) add a jazz tune, and (iii) take influences from the Smiths.

Here are certain Do’s and Dont’s when making prompts:

Don’ts:

1. Avoid trivial queries: If Google can easily answer them or if reading 1-2 website links will give the answer, it is a weak prompt. Your prompt should focus on making the LLMs work through different steps to reach the desired answer.
2. Steer clear of subjective prompts with many correct answers or are open-ended. Avoid asking for general advice or the LLM’s opinion on different subjects. The goal is to steer the LLM towards a specific or “correct” answer.
3. Using AI-generated content in any part of your prompt- be it for content or for making prompts. We will be checking every prompt through AI detection tools. If found in violation of this rule, you will be marked with an infraction, and further actions will be taken.
4. Paraphrasing is a strict no-no. Generating content with the help of AI is not the way to go; research specific topics on which you want to write the prompts. Taking content from LLMs and paraphrasing it in the hopes of fooling us will result in major complications for you as they will get flagged, and the data will be stored in your file in our database; it is highly recommended to avoid getting yourself in that fix.
5. Repetition: The LLM is being trained continuously with the prompts generated by you; so you need to keep on increasing the complexity and difficulty of the prompts with time. Avoid repeating the same kind of prompts and move on from prompts that you have already done. As the LLM will get trained on the prompts you are making from the first day itself, making the same type of prompt on the 3rd, 7th, or 20th day, will result in the LLM rejecting it, and your work will go in vain. Ensure that every prompt you write is unique.

Dos:

1. Prioritize objectivity: Good prompts will be objective, and should have limited correct answers (2-3). Achieve this by adding constraints or parameters. This is

where the importance of constraints comes into play. Rather than giving the LLM a small idea and giving it a blank canvas, specify what you are expecting from it.

2. Keep it natural: Ensure your prompts aren’t overly formal. Maintain a casual and concise tone, mirroring how users naturally ask questions. Think of it more like talking to an accomplice informally, rather than speaking with a figure of authority formally, where you are afraid to make mistakes. A natural sounding prompt can have mistakes here and there.
3. Embrace diversity; ensure variety in domain, constraints, and tone. Do not attempt two similar prompts, as doing so will result in their values diminishing as the LLM already has been trained on the first prompt, the second similar prompt wont add anything to the training. There are two major types and ten categories of prompts; attempt each of them with the goal of making the LLM work hard to give you the specific response you are looking for.
4. Avoid redundancy in prompts: Avoid doing chit-chats with the LLM; it’s a set of code defined to generate responses, and it won’t remember your personal details. Just give the necessary context, which has to be relevant to the question. Think: does the answer become more specific by the context I added? If removing the context will give the same answer, context is not needed.
5. Deliberate mistakes - Making a few spelling and punctuation mistakes adds to the natural factor. Think about how a person will normally type into the LLM; they won’t think about making the grammar and spelling perfect, as the prompt won’t be graded for the same.

Now we will look at the two different types of prompts, simple and complex.

- **Simple prompts** are questions that ask about a subject/topic by defining multiple constraints and without a context/content. They are usually defined by having less text compared to complex prompts.
- **Complex prompts** are those that ask questions based on a provided context/content. They will have a body of text or data, and the question will be asked referencing the given body of text/data.