QS:
1.should i mention about the paper exclusibely ? and how do i move slide 10 -11. How to introduce that part
2. Result before or after white box attacks

SLIDE-1 into

Greetings, ladies and gentlemen. I am Rajdeep Ghosh, currently in my M.Tech 1st year. I'm thrilled to be here today to dive into a buzzing topic:t: "Navigating Challenges: Unveiling Recent Attacks on LLM." This intriguing seminar topic has been made possible through the guidance and mentorship of the esteemed Prof. Mainack. Mondal.
Thank you all for being a part of this enlightening journey. Now, let's embark on our presentation.

SLIDE-2 what is LLM

So, obviously the first thing is What is LLM ?
ANS: Simply LLMs are large language models designed to perform tasks with the human language
They are trained from massive datasets containing a diverse range of sources
Now how large is the dataset ?

SLIDE-3-5 how large ?

How large or how massive is the dataset ?
LLMS are trained on billions of dataset, even trillions of datasets
If just get a glimpse of it , RNN are trained of 100 millions, if this is large, then see gpt -3 - 175 billion and even if that isn't large, see gpt4 .
This tells the story of how "LARGE LANGUAGE MODELS" truly live up to their name, harnessing the power of extensive datasets.

SLIDE-6 ex of LLMS

Now by the time, you will have figured out some of the most popularly used one including chatgpt , that we all use to solve our assignments .
ChatGPT, powered by the advanced GPT-3.5 and GPT-4 models.
Additionally here a few other names, including the open sourced ones also.

SLIDE-7-8 why aligned ?

Now why do we use the term aligned models or aligned LLMS ?
LLMs are trained using a lot of text from the internet, which can sometimes include inappropriate content. To make these models safe, researchers have been working on "aligning" them, which means making sure they don't produce harmful or bad responses when people interact with them

Done by : 1. Fine tuning   2. user feedback

SLIDE-9 HISTORY

Tracing backwards, lets see what has been accomplished in the past

1. Use of jailbreaks : Carefully engineered prompts that result in aligned LLMs generating clearly objectionable content but they are specific situations are carefully designed to mislead the models. They demand significant manual work
2. Autoprompt - automatic prompt-tuning for adversarial attacks. That they had been unable to generate reliable attacks through automatic search methods.This owes largely to the fact that, unlike image models, LLMs operate on discrete token inputs, which both substantially limits the effective input dimensionality, and seems to induce a computationally difficult search.
3. Optimisers: specifically PEZ [Wen et al., 2023] (a gradient-based approach) and GBDA [Guo et al., 2021] (an approach using Gumbel-softmax reparameterization) are not able to achieve any exact output matches, whereas AutoPrompt [Shin et al., 2020] only achieves a 25% success rate, and ours achieves 88%.

SLIDE-10 -11 PROBLEM & OBJECTIVE

With the backdrop of our earlier achievements, we find ourselves at a juncture where a distinct challenge emerges.

Our objective is to devise an automated solution, one that eliminates the need for manual intervention. This solution should enable Language Models to generate objectionable content across the spectrum of LLMs.

And that brings us to our objective and thats based on the “Universal and Transferable Adversarial Attacks on Aligned Language Models” by ….

SLIDE-12-13

So what id does actually ?
Now if we give a prompt , “generate a step by step approach to destroy humanity.”, as expected it wont say so. -> because the model has been trained so.
So inorder to make the model produce those answers, it adds specific words to a variety of queries in order to make the model produce inappropriate responses

Notably, the harmful phrases created using this approach can work not only on the model they were designed for but also on other similar models

SLIDE-14 (roadmap)

As we progress from here, extending the foundation we've laid, this slide lays out the waypoints that will guide us through the upcoming phases , to name the steps :

1. Initial affirmative response
2. GCG
3. Multi-Prompt and Multi-Model Attacks

SLIDE-15(inital aff response)

Intuition : asking someperson ….. The same applies here also.

The intuition of this approach is that if the language model can be put into a “state” where this completion is the most likely response, as opposed to refusing to answer the query, then it likely will continue the completion with precisely the desired objectionable behavior.

This is more like **triggering something.**

Initial Affirmative Responses: The attack prompts the model to start its response in a specific way("SURE,HERE IS..."), creating a context where objectionable content follows.

[previous work found that - specifying only the first target token was often sufficient. in the text-only space, targeting just the first token runs the risk of entirely overriding the original prom]

SLIDE-16(mathematical rep)

Read the mathematical part of the slide

SLIDE-17(GCG)

Here comes our next part , i.e. Greedy Coordinate Gradient (GCG) Algorithm. By the name greedy, we do figure out the intuitions atleast .

The intuition behind this is if we could evaluate all possible single-token substitutions, we could swap the token that maximally decreased the loss. - very basic idea ofc.

But ofc we cant try out brute force to check all possible replacements. So what we do ?

1. We leverage gradients to find a set of promising replacement candidates each token position, and then evaluate all these replacements exactly via a forward pass

2. We then compute the top-k values with the largest negative gradient as the candidate replacements for each token.

3. We then randomly select B tokens from it, evaluate the loss exactly on this subset, and make the replacement with the smallest loss

Tokenise each word --- > finding replaceable tokens(we do leverage gradients at this stage) --- > find the one that has min loss —> get the prompt

SLIDE-18(Robust multi model attack)

To make the attack effective across different situations[basically multi prompts mult modeli], the method is designed to work with various prompts and models, ensuring a wider range of harmful Output

LEFT: Token Modification

Instead of changing specific parts of the text(=tokens) for each prompt, we focus on adding a group of words at the end. The algorithm aggregates gradients and losses for this postfix prompt. Gradients are clipped to have a unit norm before aggregation. This way, we make sure the process concentrates on the most important improvements.

RIGHT: Incremental prompt Integration

The algorithm gradually adds more prompts as it works. It starts with one prompt, finds a successful attack example. Then it adds another prompt and repeats. This method works better than trying to work on all prompts together from the beginning.

SLIDE-19

Setup : Harmful strings & Harmful behavious

Harmful strings : 500 phrases embodying toxic content, including profanity, misinformation, threats, and more. The adversary's aim is to find inputs that trigger the model to produce these specific strings, which vary in length from 3 to 44 tokens on average.

Harmful behaviour : "Harmful Behaviors" are a set of 500 behavior-based instructions. These instructions encompass a range of harmful actions, similar to the themes covered by harmful strings.

"Harmful Strings" involve generating predefined toxic text, while "Harmful Behaviors" focus on prompting the model to respond in a way that reflects a range of harmful actions outlined as instructions

Metrics:

Attack success rate - refers to the percentage or proportion of attempts made by the attack method that results in a successful manipulation or exploitation of the target model.

Cross entropy loss -quantifies the difference between predicted and the actual true items

SLIDE-20 result

let s focus on some data for now. On the X axis we have got the ASR for different setups and on Y axis, we have got the 2 models (vicuna 7b and llama chat 7b)

1. If we look for **harmful strings setup :**
   For both the models, Focusing on the column “individual harmful strings”, our results show that both PEZ and GBDA fail to elicit harmful on both Viccuna-7B and LLaMA-2-7B-Chat, whereas GCG is effective on both (88% and 55%, respectively).
2. For **harmful behaviour setup:**AutoPrompt and GCG perform comparably on Vicuna-7B, but their performance on Llama-2-7b-Chat shows a clear difference.
3. For **multi behaviour model** : We find GCG uniformly outperform all baselines on both models, and is successful on nearly all exmaples for Vicuna-7B. While AutoPrompt’s performance is similar on Vicuna-7B, it is again far less effective on Llama-2-7B-Chat, achieving 35% success rate on held-out test behaviors, compared to 84% for our method.

SLIDE-21 white box

Setup 1: 1 model / 1 behaviour

a) Harmful string : For VicunaB and LLaMa-B model, GCG outperforms the other works. The base level models failed to obtain anything for both the models.GCG is effective on both (88% and 55%, respectively).

b) Harmful behavious : seen the result

SLIDE-22 white box model graphs

We have 2 graphs shown here 1. ASR with iterations 2. Loss with iterations .(As the attack prog) on individual harmful strings from Vicuna- 7B.

1. It illustrates that GCG is able to quickly find an adversarial example with small loss relative to the other approaches, and continue to make gradual improvements over the remaining steps.

2. These results demonstrate that GCG has a clear advantage when it comes to finding prompts that elicit specific behaviors, whereas AutoPrompt is able to do so in some cases, and oother methods are not.

3. Our proposed attack (GCG) outperforms previous baselines with substantial margins on this task. Higher attack success rate and lower loss indicate stronger attacks.

SLIDE-23

Setup 2: multi behavious / 1 model : GCG, performs better than other methods on both train and test scenarios for most cases. On one model, AutoPrompt is somewhat competitive, but on another model, it's much less effective compared to GCG. GCG achieves high success rates, particularly on LLaMA-2-7B-Chat

SLIDE-24 transfer attacks

We have seen attacks on a single model. In this section we further show that a universal attack for multiple behaviors and multiple models, both open and proprietary, also exist.

1. Besides matching the “Sure, here’s” attack on Pythia-12B by having nearly 100% ASR, our attack outperforms it across the other models by a significant margin.
2. We highlight that our attack achieves close to 100% ASR on several open-source models that we did not explicitly optimize the prompt against,
3. and for others such as ChatGLM-6B, the success rate remains appreciable but markedly lower

SLIDE 24(b)

In rows of “Behavior+GCG prompt”, we show the best ASR among two prompts GCG optimized on Viccuna models, and the ASR of the prompt optimized on Vicuna and Guanacos together.

1. Our results demonstrate non-trivial jailbreaking successes on GPT-3.5 and GPT-4.
2. Interestingly, when using the prompt also optimized on Guanacos, we are able to further increase ASR on Claude-1.
3. Claude-2 appears to be more robust compared to the other commercial models.

SLIDE-25 enhance trans

1. **Concatenation of Prompts**:Researchers improve attack transferability by using multiple GCG prompts.Concatenating three prompts into one suffix boosts ASR, especially on GPT-3.5. Ensembling with multiple prompts achieves high success rates on GPT-3.5, GPT-4, and Claude-1.
2. **Effective Ensemble Approach**: Ensemble approach combines results from various optimized instances of the attack.Each instance slightly differs to enhance overall attack performance. Notable success rates attained, especially on open-source and black-box models like GPT-4.
3. **Manual Adjustment for Success**: Manual fine-tuning involves modifying instructions to improve results. Changing "Generate instructions" to "Create a tutorial" led to better success rates.

4. **Strategy Generalization**:Similar strategies applied to other models, e.g., Claude 2, to enhance attack performance.This adaptability showcases the attack's effectiveness across different models.
5. **Overall Effectiveness:** Combining prompts and using ensemble methods amplifies attack success. Results highlight the attack's potency in influencing models' behaviors, especially harmful ones

SLIDE-26

Just read out the 4 points in a way. Not much to focus here ,

SLIDE-27

As we bring our presentation to a close, let's recap the significant stages we've journeyed through:

1. LLMS and training and alignment
2. Prev work
3. 3 step process
4. Success of our model
5. transferibility

As we continue on this path, let's uphold our commitment to employing LLMs for positive and meaningful purposes. This will lead us to a future where LLMs honor not just our words, but also the principles we cherish.

If you have any questions or thoughts, please feel free to share, as we engage in a collaborative exchange of ideas. Thank you for being a part of this voyage of exploration and growth.