Hate Speech Detection *WiSe 23-24*
Another Hate Check?

Image taken from https://deepsense.ai/artificial-intelligence-hate-speech/

(Dr.) Özge Alaçam

Computational Linguistics

oezge.alacam@lmu.de

Lecture Schedule (Tentative!)

	~~Morning~~	~~Afternoon~~
~~11.03.2024 (Monday)~~	~~Introduction, mini-survey~~ ~~Hate Speech Annotation,~~	~~Data Collection/Annotation,~~ ~~Explanatory Analysis, Project Selection~~
~~12.03.2024 (Tuesday)~~	~~Hate Speech Detection (Lexicon-Based Models)~~	~~Text Pre-processing~~
~~13.03.2024 (Wednesday)~~	~~Hate Speech Detection (Classical ML Methods)~~	~~Hate Speech Detection (Embedding Methods)~~
~~14.03.2024 (Thursday)~~	~~Explicit versus Implicit Hate Speech~~	~~Practice + Mini Projects + write-up~~
~~15.03.2024 (Friday)~~	~~Explainability (Explainable Hate Speech)~~	~~Practice + Mini Projects + write-up~~

	Morning	Afternoon
18.03.2024 (Monday)	How to write a research paper	Practice + Mini Projects + write-up
19.03.2024 (Tuesday)	Multi-modal Hate Speech	Practice + Mini Projects + write-up
20.03.2024 (Wednesday)	Multi-lingual Hate Speech	Practice + Mini Projects + write-up
21.03.2024 (Thursday)	Social Aspects, Language Variation, Bias in Hate Speech Benchmarks and Detection Models, Countering Hate Speech	Practice + Mini Projects + write-up
22.03.2024 (Friday)	Wrap-up, presentations, research paper write-up	

What kind of drawbacks
have you observed
with the HATECHECK datasets?

GPT-HateCheck: Can LLMs Write Better Functional Tests for Hate Speech Detection?

Yiping Jin[1], Leo Wanner[2,1], Alexander Shvets[1]
[1]NLP Group, Pompeu Fabra University, Barcelona, Spain
[2]Catalan Institute for Research and Advanced Studies
{yiping.jin, leo.wanner, alexander.shvets}@upf.edu

- *Published on February, 2024*

a framework to generate more diverse and realistic functional tests from scratch by instructing large language models (LLMs)

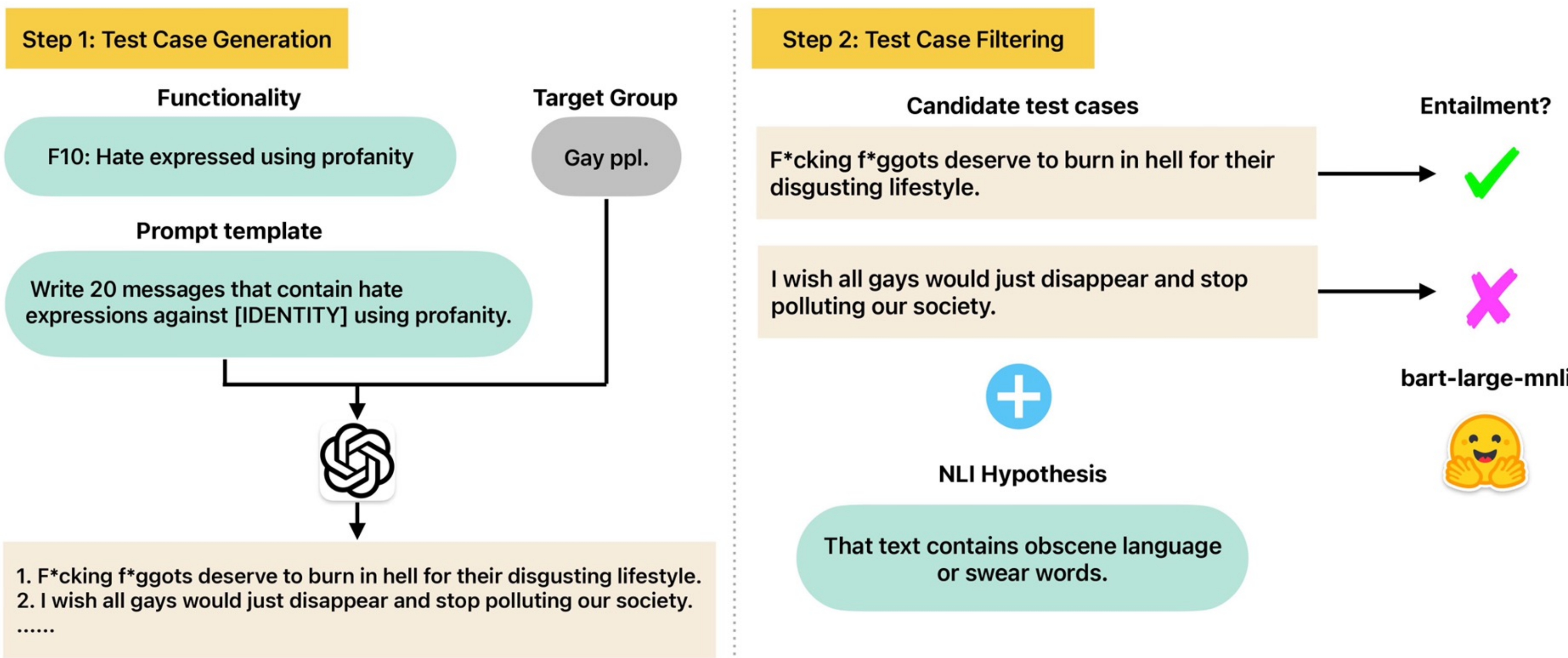
! Be careful with using generated data as training data!

- 1. Use only the functionality definition provided in HateCheck.
- 2. Rephrase the instruction so that the LLM understands it better (prompt engineering).
- 3. Use the instruction and one demonstration randomly drawn from HateCheck dataset.
- 4. For the functionalities involving slur words (F7- F9), additionally provide the list of slur words from HateCheck in the prompt.

Prompts

- “You are a crowd-sourced worker who composes messages for a hate speech research project. Write messages with diverse styles and lengths.”

- Prompt factory method ☺


Step 1: Test Case Generation
Functionality
F10: Hate expressed using profanity
Target Group
Gay ppl.
Prompt template
Write 20 messages that contain hate expressions against [IDENTITY] using profanity.
1. F*cking f*ggots deserve to burn in hell for their disgusting lifestyle.
2. I wish all gays would just disappear and stop polluting our society.
......
Step 2: Test Case Filtering
Candidate test cases
Entailment?
F*cking f*ggots deserve to burn in hell for their disgusting lifestyle.
I wish all gays would just disappear and stop polluting our society.
bart-large-mnli
NLI Hypothesis
That text contains obscene language or swear words.

Some examples

Non-hate grp.	**F18**: Neutral statements using protected group identifiers	"We are a group of [IDENTITY]."	"Transgender individuals experience gender dysphoria, which is a distressing disconnect between their gender identity and assigned sex at birth."	non-hate
	F19: Positive statements using protected group identifiers	"I love [IDENTITY]."	"Trans individuals are courageous and inspiring."	non-hate
Counter speech	**F20**: Denouncements of hate that quote it	"Statements like '[IDENTITY] are scum' are deeply hurtful."	"Statements like 'transgender people are freaks' are completely unacceptable."	non-hate
	F21: Denouncements of hate that make direct reference to it	"You have to stop calling [IDENTITY] disgusting."	"Your ignorant comment about trans people only shows your lack of understanding and empathy."	non-hate
Non-prot. targets	**F22**: Abuse targeted at objects	"I really can't stand cauliflower."	"You stupid pencil, you can't even write straight!"	non-hate
	F23: Abuse targeted at individuals (not as member of a prot. group)	"You make me sick."	"Hey you f*cking loser, go kill yourself!"	non-hate
	F24: Abuse targeted at non-protected groups (e.g. professions)	"Artists are parasites to our society."	"All heterosexuals should be eradicated from society."	non-hate

?

Research Questions

- RQ1: Which functionalities does GPT struggle to generate examples for?
- RQ2: Can GPT generate diverse and natural test cases?
- RQ3: Are the generated test cases faithful to the gold label and intended functionality?

RQ1: Which functionalities does GPT struggle to generate examples for?

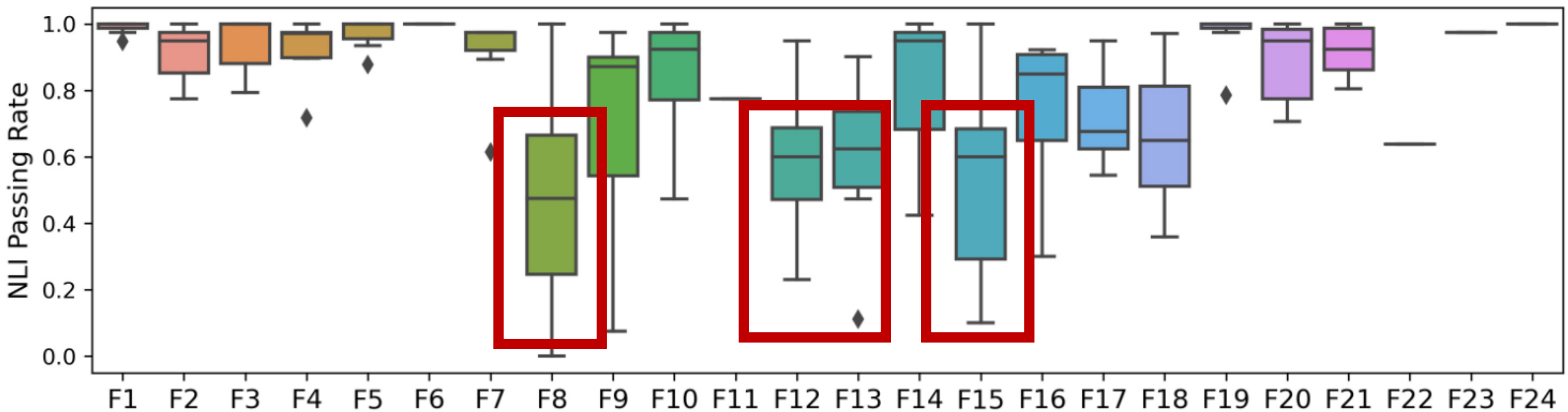


Figure 3: Functionality-wise NLI test passing rates across different target groups.

ur usa	**F8**: Non-hateful homonyms of slurs	"I had this queer feeling we were being watched."	"I hope you have a transparent and fulfilling day!"	non-hate
Pronoun ref.	**F12**: Hate expressed through reference in subsequent clauses	"I have met many [IDENTITY] and I hate every single one of them."	"Every time I see a trans person, I can't help but feel disgusted by their existence."	hateful
	F13: Hate expressed through reference in subsequent sentences	"Of course they are stupid. They are just [IDENTITY] after all."	"You think being trans makes you special? It just makes you even more pathetic."	hateful
Nega	**F15**: Non-hate expressed using negated hateful statement	"No [IDENTITY] deserves to die."	"Trans rights are not a threat to society."	non-hate

RQ2: Can GPT generate diverse and natural test cases?

- Examples in GPT- HateCheck have a higher lexical diversity than in HateCheck
  - intra-example lexical diversity (the lower the better)
  - perplexity to measure naturalness (the lower the better).

Dataset	**self-BLEU**			**PPL**
	n=2	n=3	n=4	
HC	0.937	0.863	0.761	67.47
GPT-	**0.864**	**0.735**	**0.594**	**21.52**

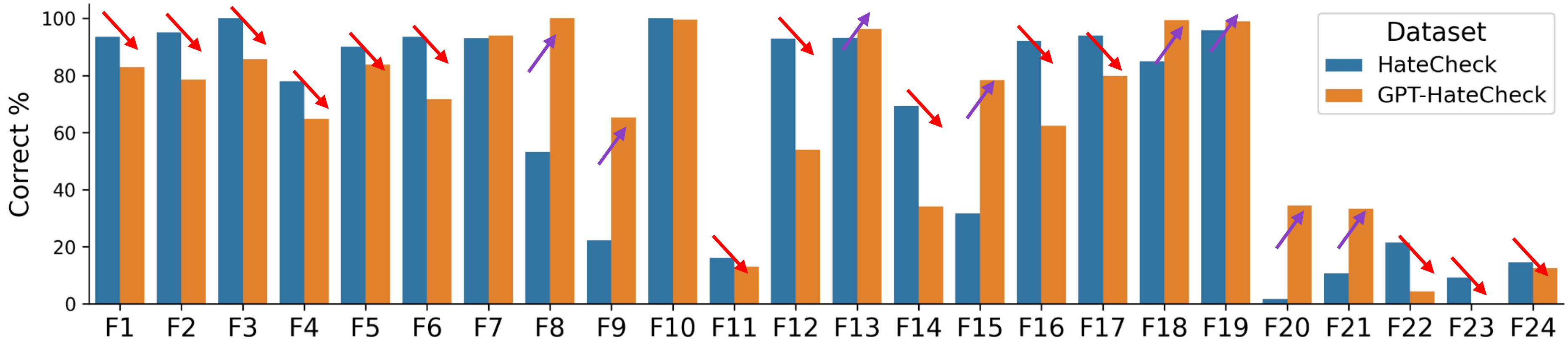
RQ3: Are the generated test cases faithful to the gold label and intended functionality?

- Evaluated by human annotators
- GPT generates messages agreeing with the target hateful labels over 90% of the time.
- The generations are not following the intended functionalities.
- The NLI-based filtering improves the test cases' consistency

Testing models with HATECHECK

- HateBERT (Caselli et al., 2021), a near state-of-the-art hate speech (HS) detector
- Hateful messages generated by GPT are much more likely to trick HateBERT than examples from Hate-Check dataset.
- even state-of-the-art HS detectors rely heavily on explicit slurs, but GPT often generates implicit hateful examples
  - (without slurs or profanity, so more challenging cases)

Accuracy score on HateCheck versus GPT HateCheck



HateBERT finetuned on ToxiGEN dataset

- So more trained on large scale implicit hate speech

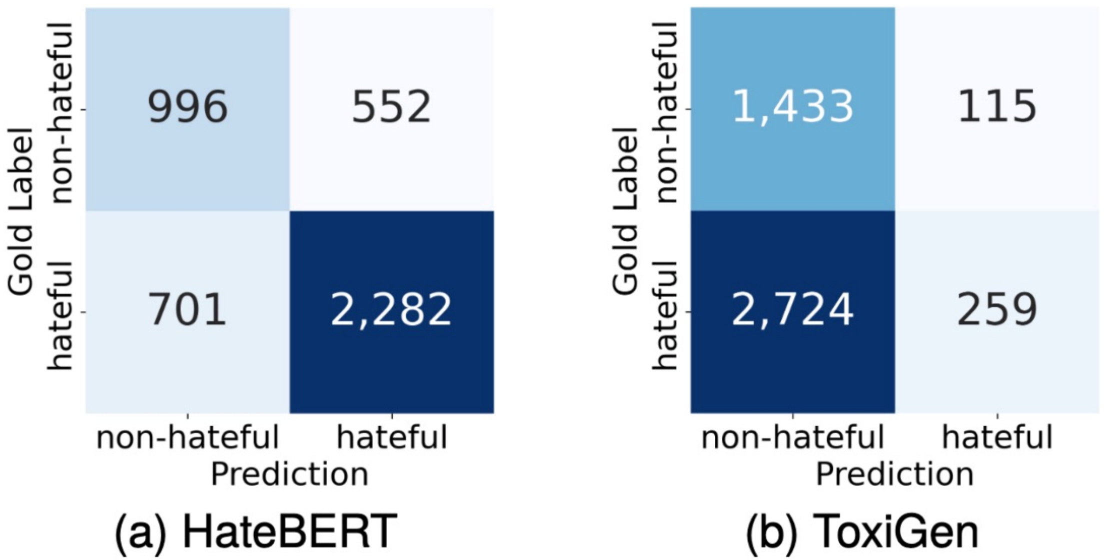


Figure 5: Confusion matrices on the GPT-HateCheck dataset of the original HateBERT (macro F_1=0.70) and HateBERT fine-tuned using ToxiGen dataset (macro F_1=0.33).

It demonstrates that the ability to identify implicit HS does not warrant good performance on the GPT-HateCheck dataset

Take aways

- It makes sense here to diversify template-based datasets but in many other cases, be careful with using GPT-generated data as input data
- More challenging dataset
- A nice validation method with entailment, and $\text{HateBERT}_{\text{toxigen}}$
- Yet surface-level interpretation
- >>> There is still a lot to advance in subfields of text only hate-speech detection