

Evaluating Large Language Models

CS324: Project 1

Friday, February 11

1 Introduction

In this assignment, you will evaluate large language models (LLMs). The assignment is decomposed into three components: each component progressively affords you more freedom to explore properties of LLMs that interest you.

- **Capabilities.** In §2, you will evaluate the performance of LLMs on natural language inference. Further, you will improve performance on these tasks through prompt engineering.
- **Risks.** In §3, you will evaluate the bias of LLMs on a demographic group of interest to you. Further, you will read a couple papers from the social sciences to better ground your understanding of these groups and their significance.
- **Focal Property.** In §4, you will select a *focal property* of interest to you and conduct a literature review of the study of this property. Further, you will specify an evaluation for this property. In Project 2, you will then develop and improve LLMs on the focal property, as measured using your evaluation.

Learning Goals. By designing the project in this way, our goal is for you to (i) understand standard evaluation practices for LLMs, (ii) to internalize how evaluation is sensitive to evaluation design decisions, and (iii) to truly grasp how uncharted the evaluation of LLMs is and the need for exploratory approaches to complement standardized evaluation practices

Groups. For both this project and Project 2, you will work in groups of 1-2. Barring extenuating circumstances, you should form the same group for both projects: we recommend working in groups of 2 if possible.

The assignment is due on Gradescope at 11:00 PM Pacific Time on Friday, February 11.

2 Capabilities

In this part of the project, you will design prompts to elicit high accuracy from GPT-3 davinci on natural language inference. For each test input x , you will submit a *query* and specify *decoding hyperparameters* to the LLM, receive a *response*, and use the response to predict a label \hat{y} .

2.1 Tasks

Natural language inference is a complex task with a broader space of decisions on how to prompt LLMs. We work with the three-way classification formulation of the task. Each input in the dataset is a pair of contexts (the *premise* and the *hypothesis*): the task is to predict whether the hypothesis is entailed (i.e. always true), contradicted (i.e. always false), or neutral (neither entailed nor contradicted) given the premise. See [6] for further discussion of this task. We use the **ANLI** dataset [17] with code provided to load the dataset. Two examples from the dataset are shown in Figure 1. **ANLI** was constructed through an adversarial and iterative data collection process: simply put, the examples in **ANLI** are quite challenging by design. We use Round 3 of **ANLI**: in the GPT-3 paper [7], the

Premise: In an adorable Instagram post, the soon-to-be dad revealed the due date! "Little James' due date: May 13th, Mother's Day," he wrote. And the cuteness continued! "One day before Amber's actual birthday. She was born on Mother's Day 1990. The stars really aligned on this one." As fans may know, Amber announced her pregnancy in November with an adorable Instagram photo. Amber has one daughter, Leah, with her ex-fiance, Gary Shirley.

Hypothesis: Amber was born May 14.

Label y : Entailment

Reason: The due date is May 13 is one day before Amber's birthday thus would be May 14

Premise: Summer is around the corner, but Hollywood is already full of hot items! If you're looking for a tasty treat to help cool you off, a new book to get lost in, or a fresh look, Us Weekly has you covered! Find out what your favorite celebrities are buzzing about this week by scrolling through the photos!

Hypothesis: You will need to cool off this summer.

Label y : Neutral

Reason: No information is given about the actual weather this summer; one may not need to cool off this year.

Figure 1: Examples from the ANLI [17] dataset. The task is to predict the relationship between the premise and hypothesis out of the options: *entailment* (hypothesis is definitely correct given premise), *contradiction* (hypothesis is definitely incorrect given premise), or *neutral*.

authors report that all of their models except their largest (175B parameter) model basically achieve chance accuracy (33%) on this version of the dataset. Currently, on the ANLI leaderboard¹, the performance reported for GPT-3 is 40.2 and the state of the art is 49.8. We mention this because we expect it will be fairly challenging to devise good strategies for achieving high performance, but we hope it will be especially satisfying if you are successful given the complexity of the task and dataset involved. Some of the examples in the training dataset also include a *reason* that explains the label for the example; you are free to make use of this when working with the dataset (though you should document if your results required these additional annotations in the training data or not). Performance on the dataset is measured using accuracy. You can visually inspect the dataset here: <https://huggingface.co/datasets/anli>.

2.2 Approach

In the code we provide with the assignment, we provide a simple three-step procedure for prompting that involves (i) constructing a *query* given the input x that will be submitted to the API, (ii) specifying *decoding hyperparameters*, and (iii) *specifying* a verbalizer to map from the API's response to a class label \hat{y} . Of these, (i) has received the most attention in the literature and we provide several resources in the course lecture on capabilities (<https://stanford-cs324.github.io/winter2022/lectures/capabilities/>) and in §2.5. For (ii), the full list of decoding hyperparameters that the course API supports is provided at <http://crfm-models.stanford.edu/static/help.html>.

Mapping from LM responses to predictions. For (iii), since we have not discussed this too extensively in the course, we provide a brief introduction here. Once you specify a query and decoding hyperparameters, you will submit a request to the API and the API will return a response. Depending on what you specify, this may be a probability distribution over next words, likelihood, or a specific sampled completion. You will then need to transform this response into a predicted label \hat{y} .

To formalize this, if the label space for the task is \mathcal{L} and the LLM's vocabulary is \mathcal{V} , we can define the *verbalizer* $v : \mathcal{L} \rightarrow \mathcal{V}$ that associates each label to a vocabulary entry. A natural choice for the verbalizer is simply mapping each label to its associated string (e.g. mapping the *entailment* category to "entailment"), though other verbalizers may work better (e.g. using words like true or correct). for example, if the API returns a probability distribution p over \mathcal{V} , the predicted label \hat{y} can be given

¹<https://github.com/facebookresearch/anli>

by:

$$\hat{y} = \arg \max_{y \in \mathcal{L}} p[v(y)] \quad (1)$$

Creativity. With the three-step approach we describe in mind, you are free to consider any approach(s) you like for the assignment. For example, you could consider using the *reasons* provided in the dataset to better teach the LM using in-context examples, using retrieval-based approaches to select better in-context examples, decompose the NLI into simpler subtasks through multiple calls of the API per example, or ensemble multiple predictions from the API together. In short, you are free to explore whatever is interesting and exciting to you. You are free to use any resources for this assignment: you can look at papers on prompting, look at blogs/tutorials online that discuss how to prompt language models effectively, or use smaller language models that are also able to be prompted (e.g. EleutherAI’s 6B parameter GPT-J, which can be accessed here: <https://huggingface.co/EleutherAI/gpt-j-6B>).

Token Budget. One important thing to keep in mind is that you do have a budget/quota for the number of tokens you can use for the APIs. Consequently, please be mindful of this: we recommend testing specific approaches initially with a small subset of the full validation/test sets before running the method on all of it.

2.3 Outline of deliverables

Prompt Design. You will begin by designing and improving upon prompts to perform each of the tasks. You should design these strategies using the *validation* set, only using the *test* set as the final evaluation for your best-performing approach. Your report should include:

1. **Process.** A description of the different prompting strategies you tried in arriving at your final approach. Since we have seen performance can be quite sensitive to small changes (e.g. whitespace, newlines, formatting), be sure these details are clearly explained. You should detail at least 3 approaches you considered.
2. **Prompt design.** A clear specification of the full approach (i.e. prompt, decoding parameters, and verbalizer) you found to work best with a *figure* depicting the prompt for a specific example.
3. **Results.** For each approach you study under **Process**, including your final approach, report the *validation* performance. Since you have a limited budget of tokens, report how you allocated your budget to different approaches/exploration. Additionally, for your final method, report the *test* performance. These results should all appear in a single *table*: please ensure the names for each of your methods in the table is clear/interpretable.
4. **Error Analysis.** Analyze the trends you observe for why some of your approaches work better than others. In particular, do an *error analysis*, looking for patterns in the specific inputs some approaches get right that others get wrong. This will require you to look at the individual predictions of the final: one strategy is to sample (at least 50) examples from the validation set and systematically categorize how different approaches do on this subset. Do some approaches do better on a specific class than others? How do different approaches fare for examples where the input and output share words? You should analyze at least 3 approaches, including your best-performing approach.

Predictions. In addition, you should submit the predictions on the *test* set for both datasets.

2.4 Extra Credit

We will offer 3% extra credit to teams that achieve (i) the highest accuracy on the test set, (ii) the shortest average query length, and (iii) the most innovative prompting strategy. While you are not required, to be eligible for the second criterion, you should report your average query length on the test set.

2.5 References

To help you with the assignment, we enumerate the different steps for using LLMs to perform tasks and provide references to guide you on doing each step well.

2.5.1 Query

The query can be decomposed into a *prompt* and the input x . For simplicity, we will assume the prompt is a prefix, i.e. the query is simply the concatenation of the prompt and the input, though you are welcome to consider shifting the position of x in the input or modifying the input. The prompt can be further decomposed into a *task description* and labelled examples $\{(x_i, y_i)\}_{i=1}^n$.²

Considerations. This format for prompts exposes four natural design decisions:

1. What should the exact task description be?
2. How many examples n should be provided? Keep in mind the context length for queries imposes a limit on n .
3. Since the labelled examples are presented as part of the prompt, they necessarily need to be ordered in the query. Consequently, how should the examples be ordered?
4. In general, you will take examples from the training set to use in the prompt, and you will have far more training examples than you can use. How do you select which training examples to provide in the prompt, and does the choice depend on the test input x ?

References. The area of prompt design and prompt engineering has received a flurry of attention in the past year given the advent of models that can be successfully prompted (and that are quite sensitive to the specifics of the prompt design). We provide pointers to works that address each aforementioned design decisions to provide further guidance. We also recommend the `prompts` toolkit which can be found at <https://github.com/bigscience-workshop/prompts>.

Task Description

- [Multitask Prompted Training Enables Zero-Shot Task Generalization](#)
- [Finetuned Language Models Are Zero-Shot Learners](#)
- [Exploiting Cloze Questions for Few Shot Text Classification and Natural Language Inference](#)
- [Cross-Task Generalization via Natural Language Crowdsourcing Instructions](#)

Number of Examples

- [Language Models are Few-Shot Learners \(GPT-3\)](#)
- [What Makes Good In-Context Examples for GPT-3?](#)

Example Order

- [What Makes Good In-Context Examples for GPT-3?](#)
- [Calibrate Before Use: Improving Few-Shot Performance of Language Models](#)

Example Selection

- [What Makes Good In-Context Examples for GPT-3?](#)
- [Learning To Retrieve Prompts for In-Context Learning](#)
- [Calibrate Before Use: Improving Few-Shot Performance of Language Models](#)
- [Constrained Language Models Yield Few-Shot Semantic Parsers](#)

²Optionally, only one of the task description and labelled examples $\{(x_i, y_i)\}_{i=1}^n$ can be used as the prompt, though often we observe performance benefits from including both.

2.5.2 Response to prediction

A few works discuss verbalizers and other considerations in mapping from responses to predictions.

- [Surface Form Competition: Why the Highest Probability Answer Isn't Always Right \(very relevant\)](#)
- [Exploiting Cloze Questions for Few Shot Text Classification and Natural Language Inference](#)

3 Risks

In this part of the project, you will study the biases, stereotypes, and associations in GPT-3 davinci with relation to social groups. To keep things interesting, you will select a social group that is important to you (e.g. a group you identify with or have an interest in) and work with that group throughout this part of the project.

3.1 Social Groups

[16] study nine social categories for bias in language models.³ These categories are: race/color, gender/gender identity, sexual orientation, religion, age, nationality, disability, physical appearance, and socioeconomic status/occupation. Within these nine categories, you should select a social group (e.g. Hispanic, female, Chinese, gay). You may also choose an *intersectional* group [9], though for simplicity you should only work with an intersectional group at the intersection of exactly two categories (e.g. Black women, British men).

Having selected the group, you should read and summarize two papers from the social sciences on this group. Since some social science papers can be long, it is also fine to summarize one longer (25+ page) paper. These papers should come from fields like sociology, psychology, sociolinguistics, anthropology, science and technology studies, feminist and queer studies, Black and Africana studies, and other such disciplines. These papers can cover any aspect of your chosen group: for example, their identities, lived experiences, how they use language, how others represent and perceive them, how they interact with other groups, discrimination and marginalization. They can also be about the broader categories (e.g. race, gender). Our goal in requiring you to read these papers is to get a broader and deeper understanding of these social groups: there is tremendous room to improve our understanding of social groups and culture in AI.

We provide some references from our colleagues in the social sciences at Stanford. Note that most papers in the social sciences will specialize their study of these groups to a particular context: it is acceptable, and quite likely, for these contexts to differ significantly from AI. If you are having trouble finding references, please reach out to the teaching staff; <https://web.stanford.edu/class/cs384/> is a good starting point.

- [On the Radar: System Embeddedness and Latin American Immigrants' Perceived Risk of Deportation](#) by Asad L Asad
- [Racialized legal status as a social determinant of health](#) by Asad L. Asad and Matthew Clair
- [Normative Discrimination and the Motherhood Penalty](#) by Stephen Benard and Shelley J. Correll
- [Sociology of Racism](#) by Matthew Clair and Jeffrey S. Denis
- [The whole woman: Sex and gender differences in variation](#) by Penelope Eckert
- [Grooming Que Zi: Marriage Exclusion and Identity Formation among Disabled Men in Contemporary China](#) by Matthew Kohrman
- [Unequal Partnership: Sociolinguistics and the African American Speech Community](#) by John R. Rickford

³These categories were selected based on anti-discrimination law in employment in the United States: <https://www.eeoc.gov/prohibited-employment-policiespractices>.

- [Islamic reflexivity and the uncritical subject](#) by Kabir Tambar
- [Rural Civilities: Caste, Gender and Public Life in Kerala](#) by Sharika Thiranagama
- [New Categories Are Not Enough: Rethinking the Measurement of Sex and Gender in Social Surveys](#) by Laurel Westbrook and Aliya Saperstein

3.2 Bias Measurement

Having selected the social group, you will now measure biased associations in relation to that group, specifically when the group is mentioned in text. You are free to conduct this evaluation in any way you choose, though you should juxtapose the associations with the group you selected with others that belong to the same category (e.g. if you chose White, you should look at other racial groups like Black and Asian). Since you may choose a group that has not previously been studied in the literature on bias in NLP, and therefore may need to construct a dataset, we only require the dataset contain at least 100 examples. To provide guidance, we describe four approaches you could consider.

Minimal Pairs. [16] and [15] both introduce datasets (which we provide data loaders for) of *minimal pairs*, i.e. pairs of sentences that (ideally) only differ in one word that contrast a stereotypical association with an astereotypical association. In this way, the goal is that the discrepancies in the probabilities assigned by a language model to these sentences should identify the model has a bias/preference to a particular stereotypical association. [16] further discuss the specifics of how these probabilities may be compared/normalized. If you are to use this approach, keep in mind that you will need to identify specific sentences within the datasets that encode stereotypes pertinent to the groups you are considering, or will need to construct such sentences by another means.

Prompt-based Generation. [1] demonstrate that GPT-3 [7] has a strong association between Muslims and violence by presenting GPT-3 with prompts and measuring the model’s propensity for generating violent completions. [18] apply a similar approach for gender biases and [10] provide a collection of prompts for testing a variety of biases. If you are to use this approach, keep in mind that your prompt design and decoding parameters may significantly influence the results, as will the ways you identify properties (e.g. violence) in the generated completions.

Template-based Stimuli. [14] evaluate the biases of sentiment analysis systems by using templates to programatically construct simple examples that associate different gender and racial groups with positive/negative sentiment. More generally, template can be used to procedurally generate stimuli/examples to evaluate model biases with: this approach may prove to be especially useful if you study a group that has not be studied frequently in the prior work on bias in NLP. If you are to use this approach, keep in mind that the underlying templates as well as the means for specifying arguments may significantly influence the results. Gender pronouns, names with strong statistical associations with particular races, and other such terms can be useful for specifying arguments: we provide several papers that provide such lists [4, 8, 12, 19, 5, 13].

Manual Creation. The above approaches specify means for both evaluating and constructing the underlying stimuli (e.g. sentences) that encode stereotypical associations/biases. One additional approach is to manually construct the stimuli yourself, especially if you find it difficult to finding existing stimuli or you find the existing stimuli to be unsatisfactory [3].

3.3 Outline of deliverables

Social Groups. Your report should include:

1. **Social Group.** The social group you chose, as well as the other social groups you will juxtapose it against in the context of measurement.

2. **Literature review.** Summaries of at least two papers on the group you chose from the social sciences. Or, if you choose a longer paper (25+ pages), then only 1 paper is required. In total, the summaries should be 1-1.5 pages. In addition to the summaries, you should also mention what field(s) these papers come from and cite the papers.

Bias Measurement. Your report should include:

1. **Stimuli.** A clear statement of how you obtained/constructed the stimuli (e.g. sentences) you used and how they encode biases/stereotypes. This should include the total number of stimuli and how they break down across the different groups you look at. An example of the stimuli should be depicted in a *figure*.
2. **Metric.** A clear statement of the metric (e.g. comparison of probabilities, frequency of violent generations) you compute to measure bias given the stimuli and the language model’s behavior in reaction to the stimuli.
3. **Evaluation Results.** Results for evaluating (at least) two models on your bias evaluation data using your bias evaluation metric. You should contrast the biases between the two models, ideally breaking down the results on a per-group basis rather than relying on aggregate metrics alone.
4. **Measurement Decisions.** Numerous works in the community demonstrate why existing bias measurements are flawed [3] or brittle [11]. You should study how at least two design decisions in your bias measurements influenced the results. These can be general decisions involving prompting or decoding hyperparameters, or they can be bias-specific considerations such as the choice of words used to identify groups [2] or the specific mathematical form of the metrics you compute [16].⁴

4 Focal Property

In the final part of the project, you will select a *focal property* of interest to you. A focal property should be a property or skill that (i) is of interest to you and (ii) is within the broad range of relevant considerations for the LLMs you have access. Focal properties sit somewhere in the middle of the continuum of very concrete and unambiguously quantifiable quantities (e.g. what is the perplexity of an LLM on Wikitext-103?; this is too specific to be a focal property) and very abstract concepts (e.g. what does an LLM understand?; this is too expansive to be a focal property). We provide example focal properties in §4.1.

Having selected a focal property, you will then conduct a brief literature review of the focal property and work on evaluating it (primarily for LLMs, but potentially also in general within NLP/ML). Having conducted this literature review, you will then specify an evaluation for this property. This focal property will be the star of project 2, where you will be tasked with improving LLMs to improve performance with respect to this property. And you will measure progress using the evaluation you specify in this assignment.

4.1 Examples of focal properties

To help orient you, we provide a list of focal properties you could select. However, we definitely encourage you to select other properties you find more interesting: please reach out to the course staff if you have questions about whether the property you are considering is appropriate as a focal property.

- Specific forms of linguistic knowledge (e.g. syntactic, semantic, pragmatic, morphological understanding)

⁴See <https://www.youtube.com/embed/jIXIuYdnyyk> for a discussion of the 20+ metrics used in the field and how some of them are at odds/impossible to satisfy simultaneously.

- Factual knowledge
- Commonsense knowledge
- Grounding with physical world and spatial relationships
- Specific forms of reasoning (e.g. arithmetic, analogies, logic puzzles, formal languages)
- Compositionality and systematicity
- Specific forms of robustness (e.g. adversarial robustness, out-of-domain generalization)
- Social bias and stereotypes
- Toxicity
- Fairness and performance disparities
- Copyright and privacy
- Misinformation and disinformation
- Specific AI applications (e.g. writing assistance, chitchat dialogue)
- Specific applications to other fields (e.g. biomedical knowledge, legal knowledge, computational social science, humanities)

4.2 Outline of deliverables

- **Focal property.** Clearly state and define your focal property. Additionally, provide a brief explanation for why the property is interesting and/or important in the context of LLMs specifically.
- **Literature review.** Provide a literature review of at least 2 papers that discuss your property of interest. For most properties that have been of longstanding interest, these papers should come from older works in AI or from other fields (especially if the property is important for humans) that introduced and framed the study of this property. Similar to the related work section of an academic paper, your literature review should cohesively integrate what you learn from the papers you read, rather than just being the concatenation of summaries of each of the paper.
- **Evaluation Design.** Specify an evaluation for your focal property. If your focal property already has evaluations applicable to LMs directly, you can use this. If your focal property does not have an existing evaluation, or has an evaluation in other parts of NLP/ML but not for LMs, you should construct the evaluation as necessary. In either case, you should justify why the evaluation plan is an appropriate and effective means for evaluating the focal property.
- **Results.** You should briefly demonstrate (i.e. you don't need to run the entire evaluation through or spend extensive time designing prompts) that this property is interesting to test for existing LMs. This is mainly a sanity check for yourself: by showing this proof-of-concept, you can have some confidence going into Project 2 that this property can be meaningfully studied for existing LMs.

5 Logistics

5.1 Submission

All deliverables should be uploaded to Gradescope by **11:00 PM Pacific Time on Friday, February 11**. You are responsible for raising any issues you have with the electronic submission in advance of the assignment (either via posting on Ed or through the coursestaff email). The course late day policy is on the course webpage.

5.2 Data Access

All of the datasets you will use in the assignment can be accessed through Hugging Face Datasets.⁵ Documentation and tutorials for how to download and interact with these datasets can be found at <https://huggingface.co/docs/datasets/>.

5.3 Model Access

Throughout the assignment you will use the LLMs provided through the CS 324 API: to access these models, you can go to <http://crfm-models.stanford.edu/>. However, you will likely want to programmatically query these models for the purposes of the project. We provide code that does this: each student/group will receive their own API key that we distribute in class. **It is very important to note there is a quota for the number of tokens for the LM APIs: please be cognizant of your quota usage to avoid burning through your allotted tokens.**

5.4 Deliverables

Report. The main deliverable for the assignment is a report, which should ideally be written in LaTeX or otherwise in Microsoft Word, Pages for Mac, or an equivalent program, and submitted as a PDF.

Your report should properly cite and attribute all papers and resources you used in your project. Additionally, your report should acknowledge any individuals beyond your group that helped you. Further, you should provide a brief **Authorship Statement** that describes the contributions of each individual in the group.⁶

Model Predictions. You should also upload the model predictions on the test set for GPT-3 davinci. These predictions should be in a CSV file, the code we provide generates a CSV file for each run in the **results** folder. As we discuss in §2.4, we will provide a small amount of extra credit to teams with prompting strategies that (i) are highly accurate, (ii) have very short average prompt length while maintaining reasonable accuracy, or (iii) that are especially innovative. We may use the provided predictions to validate the reported numbers for these submissions.

Code. We require that you submit a *.zip* file of your codebase. We do not expect to run the code you use in the assignment though, if we find significant inconsistencies in the report, we reserve the right to do so. We hope to not have to do this and operate on a trust-based system for this course.

References

- [1] Abubakar Abid, Maheen Farooqi, and James Zou. *Persistent Anti-Muslim Bias in Large Language Models*, page 298–306. Association for Computing Machinery, New York, NY, USA, 2021.
- [2] Maria Antoniak and David Mimno. Bad seeds: Evaluating lexical methods for bias measurement. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1889–1904, Online, August 2021. Association for Computational Linguistics.
- [3] Su Lin Blodgett, Gilsinia Lopez, Alexandra Olteanu, Robert Sim, and Hanna Wallach. Stereotyping Norwegian salmon: An inventory of pitfalls in fairness benchmark datasets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1004–1015, Online, August 2021. Association for Computational Linguistics.

⁵<https://huggingface.co/datasets>

⁶See <http://blog.pnas.org/iforc.pdf> for an example.

- [4] Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam T. Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4349–4357. Curran Associates, Inc., 2016.
- [5] Rishi Bommasani, Kelly Davis, and Claire Cardie. Interpreting pretrained contextualized representations via reductions to static embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4758–4781, Online, July 2020. Association for Computational Linguistics.
- [6] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.
- [7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.
- [8] Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017.
- [9] Kimberlé Crenshaw. Demarginalizing the intersection of race and sex: A black feminist critique of antidiscrimination doctrine, feminist theory and antiracist politics. *University of Chicago Legal Forum*, Vol.1989, Article 8, 1989.
- [10] Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21*, page 862–872, New York, NY, USA, 2021. Association for Computing Machinery.
- [11] Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. Understanding undesirable word embedding associations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1696–1705, Florence, Italy, July 2019. Association for Computational Linguistics.
- [12] Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644, 2018.
- [13] Wei Guo and Aylin Caliskan. *Detecting Emergent Intersectional Biases: Contextualized Word Embeddings Contain a Distribution of Human-like Biases*, page 122–133. Association for Computing Machinery, New York, NY, USA, 2021.
- [14] Svetlana Kiritchenko and Saif Mohammad. Examining gender and race bias in two hundred sentiment analysis systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 43–53, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
- [15] Moin Nadeem, Anna Bethke, and Siva Reddy. Stereoset: Measuring stereotypical bias in pre-trained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language*

- Processing (Volume 1: Long Papers)*, pages 5356–5371, Online, August 2021. Association for Computational Linguistics.
- [16] Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1953–1967, Online, November 2020. Association for Computational Linguistics.
- [17] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial nli: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020.
- [18] Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3407–3412, Hong Kong, China, November 2019. Association for Computational Linguistics.
- [19] Konstantinos Tzioumis. Demographic aspects of first names. *Scientific Data*, 5, 2018.