Towards Evaluating Creativity in Language Project Plan

Matey Krastev

m.krastev.19@abdn.ac.uk

Department of Computing Science,
University of Aberdeen, Aberdeen AB24 3UE, UK

Introduction

Large language models (LLMs) are probabilistic models of language widely used for most tasks in the field of Natural Language Processing (NLP), ranging from machine translation, text classification, sentiment analysis, auto-completion, error correction, or even simple dialogue communication. However, because, fundamentally, LLMs operate on probabilities, a lot of the applications utilizing them tend to struggle with generating logically coherent novel sequences.

Large language models are usually trained on enormous language corpora, mined from books and articles[1], social media posts[2], or otherwise internet crawls[3]. Therefore, the underlying assumption would be that, in their attempt to replicate language, they could find some success at least in terms of generating basic structure in creative fields of work such as writing code[4] or screenplays[5], among others.

While LLMs display convincingly human-like text in such areas, we instead intend to examine their ability to produce specifically *creative* text. By its very nature, creativity tends to be a subjective matter, thus, we aim to identify specific aspects of creativity that can be more commonly found within creative text, such as poetry or short stories, as opposed to other less creatively-oriented genres, e.g. news articles or user manuals. Having identified specific linguistic markers within human-written creative texts, we can then begin to evaluate the extent to which LLMs exhibit these properties in generated text. This in turn has the potential to inform techniques to adapt existing text-generation models for applications involving creative language. Therefore, we seek to compile and produce a software package that can efficiently and accurately represent and evaluate linguistic creativity.

Goals

We set out to investigate specific markers defining or correlating with conventional creativity in language. As some aspects of creativity have been investigated by research disciplines such as the field of psycholinguistics, we seek to filter and compile a set of measures that have been shown to correlate more highly with creative texts in the literature. Following that, we aim to evaluate the extent to which current LLMs can provide insights into capturing creative elements or patterns within writing. Finally, if time allows, we may use these findings to explore how natural language generation can be adapted to produce texts exhibiting more human-like levels of the creative attributes we have discovered. These can be summarised by the following three questions:

- Can psycho-linguistically motivated measures (that is, the explored metrics) successfully characterise creative properties in language?
- Can a machine learning approach be adapted for evaluating creativity in natural language?
- Can we influence a subset of current LLMs to exhibit more creative traits as defined by our creativity metric via some conventional approaches such as hyperparameter tuning or varying decoding strategies?

For the first two, we develop a suite of benchmarks we shall distribute and report the results of. For the third question, we plan to train a model maximising performance on the developed benchmarks. We will then report our findings on the best-performing models and share the architecture.

Methodology

A major part of the project will be spent on research and iterative development of the metrics and benchmarks. This is a process that requires analysing and compiling resources on creative aspects of writing and language overall. As part of addressing question two, we plan to train and evaluate a classifier based on these features. Another major factor will be the development time spent on measuring the creativity of existing language models. We choose to utilise the Transformer-based machine learning architecture[6] for some of the metrics we shall develop.

Dataset Collection

The initial steps consist dataset collection and storage. In order to perform larger-scale testing, we require clean and structured data. Therefore, we will: prepare and clean the data; apply normalisation techniques such as unknown word discounting and filtering filler words; apply parts of speech (POS) tagging via tools such as the NLTK package for Python[7].

Datasets to be used include: the ROC and Cloze Test dataset[8]; the Project Gutenberg corpus of 50,000 books[1], and the Lexical Database for English “WordNet” [9]. More potential resources may be used if the scope of the project allows.

Metric Compilation and Creation

Some groundwork has already been put forth in the field of creativity research (e.g., patterns of part of speech use, use of concrete vs. abstract concepts [10], patterns of information density[11], use of complex vs simple vocabulary [12]), however, no effort has been done to systematically compile and create benchmarks. We set out to collect, filter and produce a set of such benchmarks. We choose to implement those in the Python programming language as a command-line utility with the intent of producing a reusable package for evaluating creativity in texts. Of course, we reserve some time to test and debug the benchmarks to ensure production-level quality.

Machine Learning Approaches

Later stages of the project will require the use of machine learning frameworks. The currently preferred approach is using high-level abstractions in the Python programming language with bindings for highly-optimized parallelisable data structures implemented in commonly fast programming languages such as C/C++. Thus, we opt to apply the PyTorch library[13], which implements such GPU-accelerated machine learning development, wherever high-performance computing is desired.

We plan to make use of existing pre-trained LLMs such as GPT[14] and OPT[15] to evaluate the extent to which they can generate human-like creative text. Optionally, in subsequent experiments we also plan to explore the extent to which varying parameters during generation can induce more human-like patterns.

Resources Required

A mid- to high-end personal computer would be required for prototyping the metric, as well as building the interface and tools to interact with the system. To develop the creativity measures, we require a sizable and clean dataset as outlined above. Furthermore, a high-performance computing (HPC) cluster will be needed for utilising the large language models which have been open-sourced for research and general purposes. Python will be our programming language of choice.

An ethics review will not be required, as the project does not deal with human subjects.

Risk Assessment

The only major risks that threaten the successful execution of the project are schedule-based. The project is fundamentally research-intensive. Currently, we have mitigated this risk by carefully considering a schedule and goals outlined week by week. The schedule takes into account the author's capabilities. Frequent meetings with the project supervisor serve to check on the progress achieved in the past week, as well as to clarify the deliverables for the upcoming one. Through strict adherence to the timetable outlined below, we minimise the risks against the successful execution of the project.

Additionally, we acknowledge the scope-related risks. Depending on the project load and the progress from week to week, we run the risk of lagging behind schedule for the final task of creative text generation: as training a model tends to take a long time even on an HPC cluster, we need to be mindful of the limits the HPC cluster itself puts forward on usage. As an external factor, we may mitigate the risk by backing up the progress made in training the model, for example. Another risk related to the usage of the HPC cluster, albeit minor, could be an experienced heavy load as other students and staff may attempt to run computations during the same time, which could slow down the training time more. Still, this final task is detached from the key goals of the project, so even in the worst case, failure to accomplish it does not endanger the overall completion of the project.

Timetable

Outlined in Figure 1 is a brief overview of the expected deliverables for each week until the final deadline for the project.

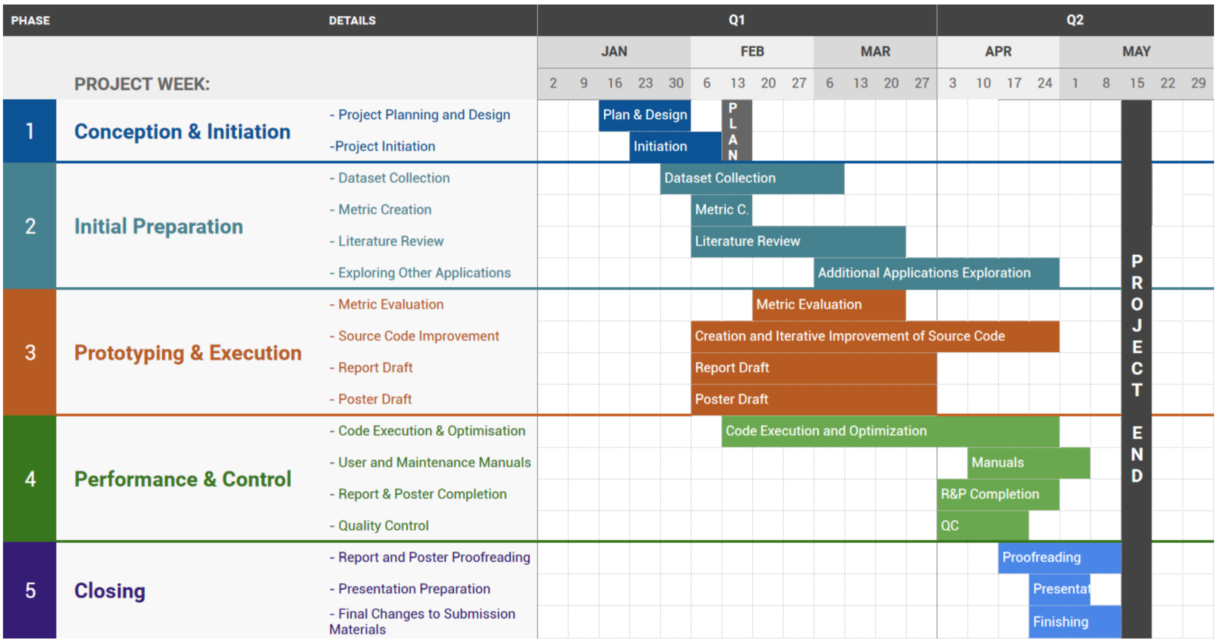

Figure 1: Timetable for Main Project Activities

References

[1] Martin Gerlach and Francesc Font-Clos. A standardized Project Gutenberg corpus for statistical analysis of natural language and quantitative linguistics. *CoRR*, abs/1812.08092, 2018.

[2] Leon Derczynski, Kalina Bontcheva, and Ian Roberts. Broad Twitter corpus: A diverse named entity recognition resource. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[3] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027, 2021.

[4] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021.

[5] Piotr Mirowski, Kory W. Mathewson, Jaylen Pittman, and Richard Evans. Co-Writing Screenplays and Theatre Scripts with Language Models: An Evaluation by Industry Professionals, September 2022. arXiv:2209.14958 [cs].

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need, December 2017. arXiv:1706.03762 [cs].

[7] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit.* O'Reilly, Beijing, 2009.

[8] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California, June 2016. Association for Computational Linguistics.

[9] Princeton University. About WordNet, 2010.

[10] Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46:904–911, 2014.

[11] Mario Giulianelli and Raquel Fernández. Analysing human strategies of information transmission as a function of discourse context. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 647–660, 2021.

[12] Victor Kuperman, Hans Stadthagen-Gonzalez, and Marc Brysbaert. Age-of-acquisition ratings for 30,000 english words. *Behavior research methods*, 44:978–990, 2012.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[14] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher 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, July 2020. arXiv:2005.14165 [cs].

[15] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open Pre-trained Transformer Language Models, June 2022. arXiv:2205.01068 [cs].