# Quote Generation in Indie Languages

Project Report - CSE582 Natural Language Processing

Ajay Narayanan Sridhar
Department - EECS
Pennsylvania State University
State College, PA, USA
afs6372@psu.edu

Rishu Verma
Department - EECS
Pennsylvania State University
State College, PA, USA
rfv5129@psu.edu

## ABSTRACT

Quote generation is an NLG task of generating coherent and meaningful words of wisdom, virtue, love, etc. In this work, we investigate the potential of generating coherent and meaningful quotes in languages that have not been traditionally employed in the realm of text generation, specifically, Tamil and Punjabi. To accomplish this, we gathered a corpus of quotes and fine-tuned a GPT-2 language model (LM), which was pre-trained on the Indie dataset, to generate fresh quotes. Our proposed methodology achieves a ROUGE-L and BERTScore of 0.201 and 0.85 in the Tamil testing dataset, and a ROUGE-L and BERTScore of 0.234 and 0.82 in Punjabi testing dataset, demonstrating that LMs can generate lucid and well-formed quotes in multiple Indie languages. Our findings hold implications for the future development of text-generation tools that can cater to a wider range of languages and cultures, and the expansion of the boundaries of the discipline.

## 1 INTRODUCTION

Quote generation is an important task in Natural Language Generation which deals with generating a coherent and profound quote. An example of a quote would be “Opportunities don't happen, you create them.”, which was penned by Chris Grosser. Quote generation task has attracted attention from the research community [1] due to its applications in the education and entertainment sectors. There has been significant progress made on this task in the English language. However, there is still a gap in the advancement of quote generation in English and other Indie languages. Quote generation in Indie languages involves several challenges such as a lack of high-quality quote data points and a lack of a vast amount of pre-training corpus.

In our task of quote generation, we focus on two Indie languages, namely, Punjabi and Tamil. Punjabi is an Indo-Aryan language of the Punjab region of Pakistan and India. It has approximately 113 million native speakers. Punjabi is the eleventh most widely spoken language in India. Tamil is one of the longest-surviving classical languages of India. It has approximately 85 million native speakers.

To tackle the issue of a lack of quote-related datasets in Indie languages, we explore two different approaches.

- For Tamil, we scrape quotes from a famous classic Tamil literature called Thirukkural, which consists of 1330 short couplets, or Kurals, of seven words each. The text is divided into three books with aphoristic teachings on virtue (**aram**), wealth (***porul***), and love (***inbam***). Its authorship is traditionally attributed to Valluvar, also known in full as Thiruvalluvar.

**Thirukkural:**
அகர முதல எழுத்தெல்லாம் ஆதி பகவன் முதற்றே உலகு.

**English meaning:**
*“As the letter A is the first of all letters, the eternal God is the first in the world.”*

- For Punjabi, we experiment with translating a widely used English Quotes-500k [2] dataset to Punjabi. For our experiments, we limit the number of records to 50k.

**Punjabi Quote:**
ਇੱਕ ਦੋਸਤ ਉਹ ਹੁੰਦਾ ਹੈ ਜੋ ਤੁਹਾਡੇ ਬਾਰੇ ਸਭ ਕੁਝ ਜਾਣਦਾ ਹੈ ਅਤੇ ਫਿਰ ਵੀ ਤੁਹਾਨੂੰ ਪਿਆਰ ਕਰਦਾ ਹੈ

**Equivalent English Quote:**
*“A friend is someone who knows all about you and still loves you”*

We experimented with numerous models, including traditional statistical models and advanced deep learning models, to ascertain which approach would prove the most efficacious. Our initial results showed the supremacy of

deep learning models for quote generation. Thus, we went ahead with Large-scale Language models for generating quotes. We use an LLM, GPT-2 [3], which was pre-trained in various Indie languages for our task of quote generation.

Our contributions are as follows,

- We propose a GPT-2 based model for the problem of quote generation in Tamil and Punjabi.
- We explore the method of lingual transfer learning from English to Punjabi for the task of quotes generation.

## 2 RELEVANT LITERATURE

This section provides a comprehensive overview of a few articles and relevant papers we went through while researching for the project idea.

**Quote Generation by Guardian**

This article[7] is a very good read about “The Guardian” a UK-based News organization’s NLU system, where they used a combination of rule-based and machine-learning approaches to extract information from text posted. The article provides an overview of the system's architecture and key components, including the use of custom entity recognition models, dependency parsing, and named entity recognition. The article also describes how The Guardian used a combination of human annotation and active learning to improve the accuracy of their models. This article helped us understand how we can extract the information and improve the model using human annotations.

**Evaluate Text Generation Models**

In this article[8] the author has discussed the methods to evaluate the text generation models. The article discussed metrics like perplexity, BERTScore, and BLEURT. Perplexity measures how well a language model predicts a given text. The lower the perplexity score, the better the model is performing. The author explains how perplexity is calculated and how it can be used to compare different models. BERTScore and BLEURT take into account the semantic similarity between the generated text and the target text. BERTScore uses a pre-trained BERT model to compare the embeddings of the generated and target text, while BLEURT is a more flexible metric that can be trained on specific tasks and domains. Emphasis is also made on the human evaluation methods, which is still considered the gold standard for evaluating the quality of text generation and this was also emphasized by Professor in class. However, the author notes that automatic evaluation metrics can provide useful insights into the performance of a model and can be used to guide further development and optimization.

**Benchmark Datasets**

In the paper [4] the authors have tried to benchmark datasets in the Indian language on multiple text generation tasks. Almost all the tasks are implemented in most of the Indian languages but quote generation is a novel task and yet to be implemented on these benchmark datasets. The paper is indeed a great read and helps understand the benchmarking and complexity of native languages.

## 3 METHODOLOGY

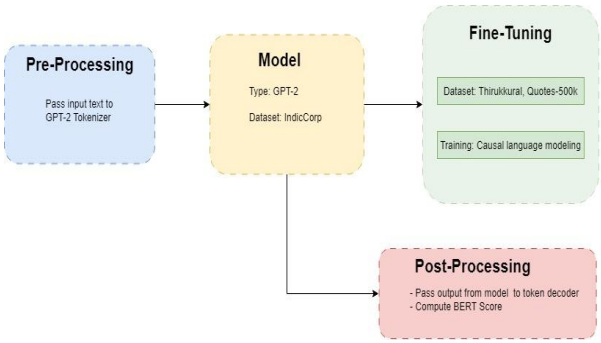


Figure 1: Illustration of our proposed pipeline. We pass the input text to a tokenizer, pass the generated tokens to the gpt-2 model, and decode the output from the model to generate a quote. The input text we pass during inference will be the starting word of the quote.

In this work, we exploit the knowledge learned from large-scale datasets and use the method of transfer learning to adapt the LM to our task. The pipeline of our proposed method is shown in Fig. 1.

### 3.1 Pre-processing

The input to our model will be either text in Punjabi or Tamil. We pre-process this input by passing it through a GPT-2 based tokenizer.

### 3.2 Language Model

We use a GPT-2 transformer model [3], which is trained on IndicCorp [4]. IndicCorp is one of the largest publicly-available corpora for Indian languages. It contains a total of 8.8 billion tokens across all 11 languages and Indian English, primarily sourced from news crawls. We

hypothesize that pre-training on this large-scale dataset will help in learning the structure of Indie languages better.

### 3.2 Post-processing/Inference

During inference, we give the starting word as an input to the model, and the model generates a token as output. We decode these tokens to formulate the generated quote.

## 4 DATASET

### 4.1 Thirukkural

Thirukkural is a famous Tamil classic literature, which consists of 1330 short couplets, or kurals, of seven words each. There are several web pages that have translations and meanings of each kural in English. We use a script to scrape this data from online sources. We split 1330 kurals into 1064 data points for training, and the remaining, 266 into the testing datasets. We show the word cloud of Thirukkurals in Fig 2.

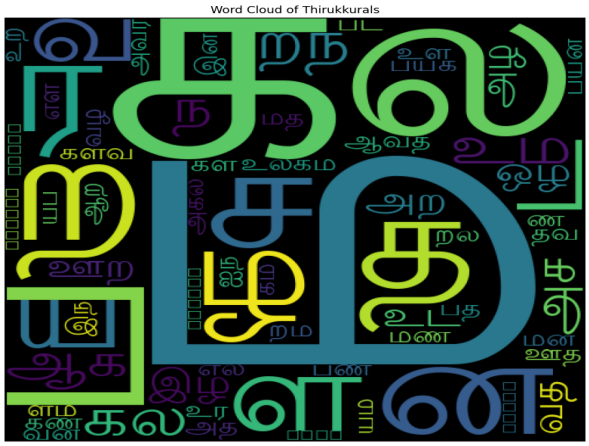


Figure 2: Word cloud for all the kurals in Thirukkural text.

### 4.2 Quotes-500k

Quotes-500k is a large-scale English quotes dataset with around 500k quotes. It contains quotes from various popular websites — Goodreads, Brainyquotes, Famousquotesandauthors, and Curatedquotes. The dataset has three features — the quote, the author of the quote, and the category tags for that quote. Examples of tags include — love, life, philosophy, motivation, family, etc. These tags help in describing the various categories that a particular quote belongs to.

For our task, we use Google's MT5 [5] model to translate English quotes to Punjabi equivalent quotes. We sample 10k data points from the Quotes-500k dataset for our task of quote generation. We split the 10k samples into 6k training samples and 4k testing samples. We refer to this converted dataset as the Punjabi Quotes-500k dataset.

## 5 METRICS

We use ROUGE-L, and BERTScore as evaluation metrics for our task of quote generation.

. .

- ROUGE is a recall-focused metric. We consider ROUGE-L which checks for Longest Common Subsequence.
- BERTScore is an automatic evaluation metric used for testing the goodness of text generation systems. Unlike existing popular methods that compute token-level syntactical similarity, BERTScore focuses on computing semantic similarity between tokens of reference and hypothesis

.

## 6 IMPLEMENTATIONS

We perform the fine-tuning of the GPT-2 model using the hugging face module. We use Adam Optimizer to optimize the model for 4 epochs with the hyper-parameters as follows: batch size = 32, betas = (0.9, 0.999), epsilon = 1e-8, initial learning rate = 5e-4. The learning rate is decayed using a cosine annealing scheduler. We also perform data cleaning on both datasets, removing newline characters, and formatting the data for easier parsing while training.

## 7 EXPERIMENTS

We perform several experiments to arrive at our final GPT-2 model.

### 7.1 LSTM model

We explored the LSTM model for the task of quote generation. We stacked 4 LSTM modules on top of each other and trained the model on the Thirukkural dataset and Punjabi Quotes-500k dataset. Tab. 1 shows evaluation metric scores on the LSTM model. We observe that the LSTM model performs poorly on this task, and hence we explore transformer architecture for the same,

| Dataset | ROGUE-L | BERTScore |
|---|---|---|
| Thirukkural | 0.01 | 0.24 |
| Punjabi | 0.02 | 0.29 |

| | | |
|---|---|---|
| Quotes-500k | | |

Table 1: Evaluation metrics for LSTM model

### 7.2 Impact of #layers of GPT's Architecture

In this experiment, we change the number of layers of GPT-2 and see the impact of scores. We start with 12 layers and increase progressively to 48, which corresponds to GPT-2.

| Layers | ROGUE-L | BERTScore |
|---|---|---|
| 12 | 0.174 | 0.75 |
| 24 | 0.182 | 0.78 |
| 36 | 0.195 | 0.82 |
| **48** | **0.201** | **0.85** |

Table 2: Evaluation metrics on Thirukkural dataset

| Layers | ROGUE-L | BERTScore |
|---|---|---|
| 12 | 0.204 | 0.78 |
| 24 | 0.212 | 0.81 |
| 36 | 0.221 | 0.84 |
| **48** | **0.234** | **0.89** |

Table 3: Evaluation metrics on Punjabi Quotes-500k dataset

Based on the results from Tab 2 and 3, we choose 48 layers as the final model which correspond to GPT-2 Architecture.

### 7.3 Impact of #input words during inference

During inference, we can change the number of words given as the input to the model. Increasing the number of words will improve performance as there will be more overlap between ground truth and generated text. From Tab. 4, we can verify the above intuition empirically on the Thirukkural dataset.

| #words | ROGUE-L | BERTScore |
|---|---|---|
| 1 | 0.201 | 0.85 |
| 2 | 0.263 | 0.89 |
| **3** | **0.339** | **0.92** |

Table 4: Impact of changing the number of words considered during inference on GPT-2 for Quote generation in Tamil

Although increasing the #input words will improve the performance, we want the model to generate coherent quotes without many input words. Thus, we pass only one input word for our quote generation task.

## 7 Results & Discussions

In this section, we show qualitative and quantitative results of the quotes generated by our model. We also discuss the shortcomings and strengths of our model based on the results.

### 7.1 Qualitative Results

#### 7.1.1 Punjabi Text

When we give 'ਪੁਰਾਣਾ' as the input to our model it generated a quote as "ਪੁਰਾਣਾ ਸੋਨਾ ਹੈ" which means the "Gold is old", although this sentence is grammatically correct, but when we think of it from the point of a quote or a word of wisdom it doesn't make much sense. It just acts like an informative sentence which means that the Gold is old. But the original quote "Old is Gold" means the older the source the better it is.

#### 7.1.2 Tamil Text

Given, 'அருள்வெஃகி ஆற்றின்கண்' as input text, our model generated the following kural on the input words.

'அருள்வெஃகி ஆற்றின்கண் பார்வம் இன்னொரு கொண்டும் தீர்க்கேற்றவாதல'

The above text literally translates 'Arulveki River's eye view is unmatched by anything else'. Although the literal translation's meaning is not apparent, the meaning in Tamil is profound. One who is aiming for wealth will perish eventually compared to the one who is aiming for kindness. In this case, our model actually comes up with a coherent and profound quote.

Another example, where "பொய்யாமை" is given input text, our model generated the following kural,

"பொய்யாமை பொய்யாமை. ஆற்றின் அறம்பிற்பாடு. இன்றுவாழ் வேந்தர் க"

The above text reuses the word 'பொய்யாமை' twice and the sentence is not very coherent. Another shortcoming that can be fixed by post-processing is random full-stops in between words.

### 7.2 Quantitative Results

In Tab 5, we show the quantitative results of our model on the testing portions of the Thirukkural and Punjabi Quotes-500k dataset.

| Dataset | ROGUE-L | BERTScore |
|---|---|---|
| Thirukkural | 0.201 | 0.85 |
| Punjabi Quotes-500k | 0.234 | 0.89 |

Table 5: Evaluation metrics of our model on testing datasets

## 8 CHALLENGES AND FUTURE SCOPE

Based on the results and observation from the implementations made we can extend this implementation to other native languages like Malayalam, Odia, Nepali, etc. Moreover, currently, the data on which we trained the model was limited and the main limitation was that if we extend the dataset further by translation from English (readily available) the dataset generated was noisy (as it was grammatically correct but intuitively it was not correct) i.e. it lacked quality. Hence, we can also work on finding more efficient ways to generate the datasets.

This model can further be extended to conditional quote generation like, the user can ask the model to generate a quote related to happiness, or sadness, without any bias, etc. Then we can generate our own new evaluation metric by creating another model which determines how much the generated quote is associated with the input condition more like zero-shot classification in the pipeline library from the hugging face.

## 9 ACKNOWLEDGEMENT

We extend our sincere gratitude to Professor Wenpeng Yin for their invaluable assistance and guidance throughout the semester and this project. Their support has played an instrumental role in our academic progress and development, and we greatly appreciate their dedication to our education. We also want to thank the Teaching Assistant, Shravya Chillamcherla, for her unwavering support and assistance throughout the course. Her expertise and dedication have been invaluable in helping us navigate the course material. We are grateful for their guidance and support and express our heartfelt thanks to both of them.

## REFERENCES

[1] Aditi Sharma, Divya Gupta, and Mukesh Kumar. 2022. Context-Based Quote Generation from Images. In Proceedings of 2nd International Conference on Artificial Intelligence: Advances and Applications: ICAIAA 2021. Springer, 815–823.

[2] Shivali Goel, Rishi Madhok, and Shweta Garg. 2018. Proposing contextually relevant quotes for images. In Advances in Information Retrieval: 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings 40. Springer, 591–597.

[3] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and others. 2019. Language models are unsupervised multitask learners. OpenAI blog 1, 8 (2019), 9.

[4] Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks, and pre-trained multilingual language models for Indian languages. In Findings of the Association for Computational Linguistics: EMNLP 2020. 4948–4961

[5] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. (2021).

[6] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL '02). Association for Computational Linguistics, USA, 311–318. https://doi.org/10.3115/1073083.1073135

[7] https://explosion.ai/blog/guardian

[8] Evaluate Text Generation