



Text Generation Tool for Writing Assistance using Transformer

Shubham Baid^{1*}, Rupera Jalay Rajeshbhai², Shubham Agarwal³, Kiran M⁴

^{1,2,3,4}Computer Science and Engineering, REVA University

Email: ¹r17cs391@cit.reva.edu.in, ²r17cs176@cit.reva.edu.in, ³r17cs390@cit.reva.edu.in,

⁴kiran.m@reva.edu.in

*Corresponding author's E-mail: r17cs391@cit.reva.edu.in

Article History	Abstract
Received: 06 June 2023 Revised: 05 Sept 2023 Accepted: 30 Nov 2023	<i>Creative writing is a challenge, and everyone faces at some point of time. The struggle of writing with the help of web is really high because of the fact that even though we try, we cannot really bypass the plagiarism. Hence, we created this tool named, Hailey – Text Generation Tool for Writing Assistance using Transformer. It leverages the power of GPT-2 Transformer and generate completion texts conditionally. This tool helps users by finishing their sentences without conditions. Users just have to enter their desired sentences and the tool will complete the paragraph by adding follow up sentences and words. This tool will be really helpful when users want to generate content in a short period of time or for getting new ideas for framing the article, stories and so on.</i>
CC License CC-BY-NC-SA 4.0	Keywords: Text Generation, Transformers, Creative Writing, GPT-2

1. Introduction

We live in India where, most of the people are cannot be able to read the details of the medicine we have to provide to someone who is sick, the menu at any eatery/canteen or warning signs on the roads; a place where we cannot be able to understand the government regulations provided in our driving license, passport or in ITR filing. This is the hiatus that huge number of Indians face simply for the reason that committee prefer English language. The statistics shows that due to various reason only 30% of the population knows acceptable English, so that they can rely on their skills to survive in any country or organization where they are forced to use only English. Since English is pretty much a necessity nowadays the awareness of writing and reading aptitude is really important, being a 125M population and one of the largest economies in the world, it's not really impressive that only 10% of it can properly read and write in English. With a recent survey and stats by eMarketer in April 2018, it has been found that by 2022, 36.2% of the Indians will have access to smartphones. The complete objective of this tool is to help individuals who find it difficult or wants to get started with creative writing.

Literature Survey

In 2019, HuggingFace published the paper based on Transformers, Transformer architectures are used to build high-capacity models. It was possible from the pre-training to utilize the capability of this architecture for various tasks. It has been open-sourced so that it can be accessed by the machine learning community. The whole architecture is consolidated into an easy-to-use API. Few pre-trained models are also available with it for quick integration by the users. It is designed to make the workflow for researchers, product deployments and practitioners really easy. [1]

In Dec 2017, Research team from Google had proposed a modern network architecture, the Transformer, it is based entirely on the concept of making recurrent and convolutional neural networks see where to look ahead, also known as Attention Mechanism. These models yielded better results and were superior in quality while being quicker to train and easier to parallelize when experimented on couple of machine translation jobs. The model outperformed in the English-to-German translation task by scoring BLEU of 28.4 score. This prototype on the WMT-2014-English-to-French translation job, scores astonishing 41.8 BLEU score while being trained for over 3 days on 8 GPUs which is really less

as compared to some of the pre-existing literature models. Transformers are proving to be very effective on generalizing other tasks having large and limited training data. [2]

In 2018, A new statements was given by OpenAI that the language models can learn without supervision and the test was done on the dataset which has a collection of huge number of webpages. One of the tests conducted as well had a consolidated text consisting of questions and the model was given to generate the answers for the same, the model proved to score a F1 score of 55 which was trading blows with prominent baseline systems without even using the same amount of training examples. The main reason behind the success of the model is its capacity which is helping in increasing the performance. The proposed prototype GPT-2 is a huge transformer(s) having over 1.5 billion parameters and it seamlessly performs after being tested on. The generated sample(s) from the prototype shows coherent paragraphs and noticeable improvements. These observations show a clear path towards building of efficient language processing systems which can learn to perform tasks more effectively from general demonstrations. [3]

Again, in June 2019, OpenAI's team demonstrated that in order to observe the gains on these tasks, the generative pre-training should be done using unlabelled data from various sources and addition to that fine-tuning of the model should also be done after the completion of each task. Also, to differentiate from the previous approaches, the team used transformers during the fine-tuning of the model to get greater learning by not making much changes to the model architecture. The claims are made by testing and benchmarking the model on various natural language understanding datasets to get a wider view on the effectiveness. GPT2 task-agnostic prototype tops discriminatively trained prototypes which uses designs which are explicitly constructed for each job, with really high improvements on the state-of-the-art in 9 out of the 12 jobs examined. [4]

Speech is a really convenient and oldest way of information exchange between people. Many attempts have been done to develop computers which can interact vocally. Such interfaces would yield great advancements in computing and many greater benefits. Here, in this scenario, a machine be able to synthesize the text as well as provide a speech. Text-To-Speech (TTS) Synthesis is a technology in which computers takes input in a descriptive form and converts it into a speech like audio and in this case, it is converted to English speech. The implementation in this paper is done using the language JAVA with multiple external libraries. The design is pitched in the direction of offering a one-way transmission interface whereby the machine connects with the user by reciting textual document for multiple different use-cases. [5]

Using the ideas and advancements from all the above-mentioned contributions, we have leveraged the transformer architecture and the pre trained model from the Huggingface's archive and ported it to pytorch framework to make the integration seamless with the webapp. Along with the GPT2 model, we have also integrated the Text-To-Speech module by Google which is helping us get the voice output for the generated text.

2. Materials And Methods

The whole tool is divided into three modules based on functionality. First is a pretrained GPT2 transformer which is generating the text-based on the input offered by the user, second module is the flask server which is responsible for helping user communicate with the app and vice versa and finally the TTS service, which is giving voice outputs to the user.

A. Transformer

Self-supervised model pre-trained upon a very huge number of English datasets. The prototype was pre-trained on the texts which was not labelled by the developers, it generated inputs and labels from the datasets automatically which is the sole characteristic of attention network. The prototype was trained that can be able to predict the next word in given sentences.

In particular, sequences of continuous texts are given as input keeping the length of input in mind and the validation output is also of the same sequence, but they are shifted on piece of the word. The model makes use of a masking mechanism to ensure that prediction of each piece of word uses the pieces from the input sequence.

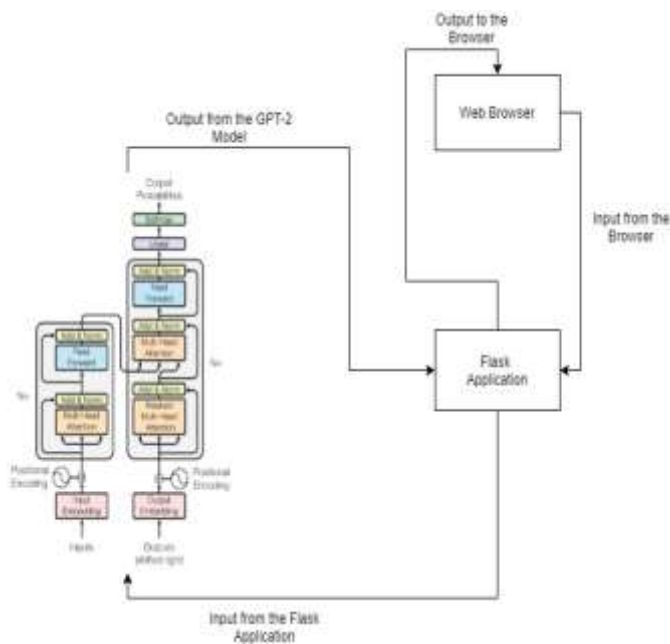


Fig. 1. Implementation Blueprint

By this approach, the prototype learns English language descriptively so that the knowledge can be used later to extract features useful for other tasks. The prototype is best at what it was pretrained for however, which is generating continuous text from a given sentence.

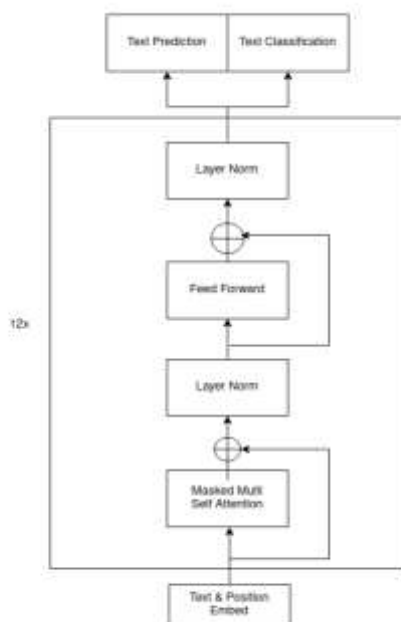


Fig. 2. Representation of Text Generation by Transformer

The sample dataset on which the model has been trained is in the form of JSON.

```
This example was too long and was cropped:
{
  "text": "\"A magazine supplement with an image of Adolf Hitler and the
```

Fig. 3. Sample WebText dataset

B. Flask

It is a web framework based on python, which is being used as a server in this tool to communicate with the transformer and the user via GUI. It is also helping with the communication with Text to Speech

module. The output generated by the transformer is stored in a file, which is later used by the flask to render to both textual output as well as the audio output.

C. TTS

The text to speech module used in this tool is GTTS. GTTS reads the text from the given input source and uses audio manipulation to generate audio output for a better interaction medium for the user.

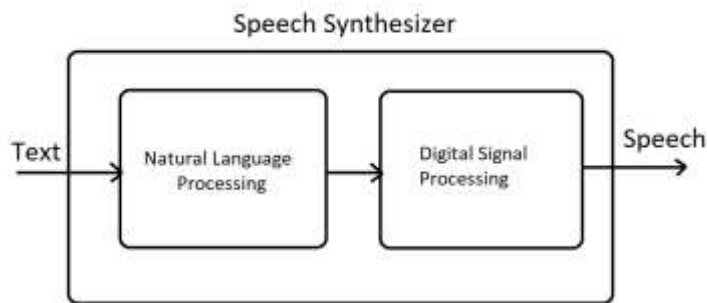


Fig. 4. The flow of TTS Synthesis

D. GUI

The GUI is kept quite subtle and easy to use with single button interface, developed using HTML and CSS. The complete pipelining of all the modules is done using python and it is made sure that the tool can be accessed through any device by keeping cross-platform interface in mind.

3. Results and Discussion

Our proposed tool managed to generate plagiarism free and error free sentences out of the input sentences. The screenshots of the tool's UI and generated output are attested here.

There is a provided text box where the user needs to enter the sentence and click the “show me” button to get the output, the same time the generated text will be shown in the below text box and the voice output is played at the same time.



Fig. 5. Hailey's Result – 1



Fig. 6. Hailey's Result – 2.

4. Conclusion

As we all know the challenges of creative writing and how time-consuming process it is. With the current results, we have the proof that this tool can tackle all the present issues and help the user with the same. Transformers have a long way to go, and possibly synthetic conditional conversation generation becomes a thing as well.

Our goal is the same, we are actively in progress to integrate speech-to-text module for seamless conversational interaction with the tool. Along with the integration of speech-to-text we are looking forward to integrating the ability to use this tool in any possible language for wider scalability.

References:

- [1] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, Alexander M. Rush, "HuggingFace's Transformers: State-of-the-art Natural Language Processing", arXiv:1910.03771v5, October 2019.
- [2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, "Attention is all you need", arXiv:1706.03762v5, 6 December 2017.
- [3] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, "Language models are unsupervised multitask learners", Semantic Scholars, Corpus ID: 160025533, 2018.
- [4] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, "Improving Language Understanding by Generative Pre-Training", Semantic Scholar, Corpus ID: 49313245, 2019.
- [5] Nwakanma Ifenayi, Oluigbo Ikenna, Okpala Izunna, "Text-To-Speech Synthesis", IJRIT, Volume 2, Issue 5, p154-159, Sung Won Lee, SeokJin Kim, Jeong Han, Kwang Eun An, Seung-Ki Ryu, Dongmahn Seo "Experiment of Image Processing Algorithm for Efficient Pothole Detection" IEEE International conference, 2019.