Domain Based Article Generator

Abstract

I will be looking into a machine learning algorithm with the primary objective being article generation; a system which will help in writing informative articles most likely to catch the reader's attention. I plan to use a combination of both BERT[1] and OpenAI – GPT[2] to address different aspects of the generator.

Research Problem

Currently, almost all software utilized to help with writing any form of literature are based on correction. After a user has input what he wanted to write,ss suggestions are given to make corrections to grammar, spellings and such.

The prime examples are the corrective predictions on Microsoft Word and Grammarly. Grammarly, the more advanced software, also allows the user to make changes the voice and change the structure of the sentences but it often misses the context of the article. It makes suggestions which can hamper the technical aspect and the actual meaning of the text.

Furthermore, these software are not trained on domain specific text, as a result it cannot identify many keywords or phrases which are required in the articles and can only give suggestions for generic mistakes in grammar.

The Neural Article Generator will look into this field from a different perspective. Its objective is to generate text and writing prompts based on a set of instructions given by the user. It will be trained on domain specific material and will learn the keywords and phrases which are frequently used in related articles.

By providing predictions at the start, it will help writers easily get overs writer block. The user will be able to choose the tone, put in main keywords they want in their article and the generator will put forward multiple writing prompts to choose from.

Literature Review

There has been a lot of work especially in the recent past in the field of text generation. OpenAI's GPT[2] is the benchmark in this field and our model will be partially based off that. However, GPT[2] is a generalized text generator i.e., not topic specific. It also has flaws such as lack of regularisation on vulgar language.

Hierarchical Neural Story Generation (Angela Fan, Mike Lewis, Yann Dauphin, 2018)[3] is another paper which has looked into a similar implementation. The paper looked into creative

systems that can build coherent and fluent passages of text about a topic. The hierarchical approach taken up by them will help us structure the text generated by the model and make it more systematic and legible.

However, most of the papers and models in this field looked into generalised text generation whereas I intend to delve into specific topic-based algorithms. Though this will make the model heavier and will require segmentation of training data, it will ultimately be beneficial to the final output of the model since texts from different fields are aimed at different audiences. As a result, the language used; even excluding keywords and such, will vary. Domain specific training will help the machine generate more engaging articles targeted at the specific audience the user desires

Training Dataset

The training dataset will be a set of 250,000+ articles on various terms which garner the most traction when a given topic or keyword is searched on Google.

Since there are a lot of topics, I will focus primarily on articles from the fields of Machine Learning, Robotics, Electronics and Software Engineering. Since these topics are often aimed at a similar crowd, the algorithm will be able to pick up on patterns on the structure of articles even better though the keyword encoding will be topic specific

A big impasse that we'll face while creating the dataset is filtering out clickbait articles. This will be a problem we face will creating the datasets. These articles often get a high number of views but they lack in depth and quality of writing. As a result, these articles will be very good for capturing keywords but will be hinderance to the sentence formation and article structuring. It'll be a challenge to filter such a huge set of articles into clickbait and quality articles without separating them manually.

Model Selection

The model will have to learn mainly two different features from the article; one being keywords and key phrases and the other being grammar, article structuring and sentence generation. Thus, it will be based on a combination of both BERT[1] and OpenAI – GPT[2] to address the different aspects.

BERT[1] is extremely adept at understanding context from sentences while GPT[2] is at the forefront for text generation. The BERT[1] layers will thus serve as the base of the model and will generate related keywords based on the input of the user and the GPT[2] based model will act as a secondary model and will output the final framed article.

Future Implementation

This article generator will continue training as it is being used. The articles it will produce will be looked at by the writer who will put in their changes or go with the generated article itself. This piece of text post-edit will be passed through the generator and thus increase future efficiency.

Once it has become very well trained on certain topics it can then be utilised to generate guides for people looking into that field. In a similar fashion it can also be utilised to answer FAQ(s) by focussing on the keywords given in the question.

References

1. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova (2019)
   BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
   arXiv:1810.04805v2 **[cs.CL]**

2. Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever (2018)
   Improving Language Understanding by Generative Pre-Training
   https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

3. Angela Fan, Mike Lewis, Yann Dauphin (2018)
   Hierarchical Neural Story Generation
   arXiv:1805.04833v1 **[cs.CL]**