Phantom Pen

Your digital ghost writer

Mobolaji Williams — Insight Data Science

Demo Video

https://youtu.be/W-UbsOGW3KQ

Relevant Code

https://github.com/mowillia/phantom_pen

Responding to Writing Needs with Machine Learning

Need #1

Move past a mental block and generate new ideas for writing

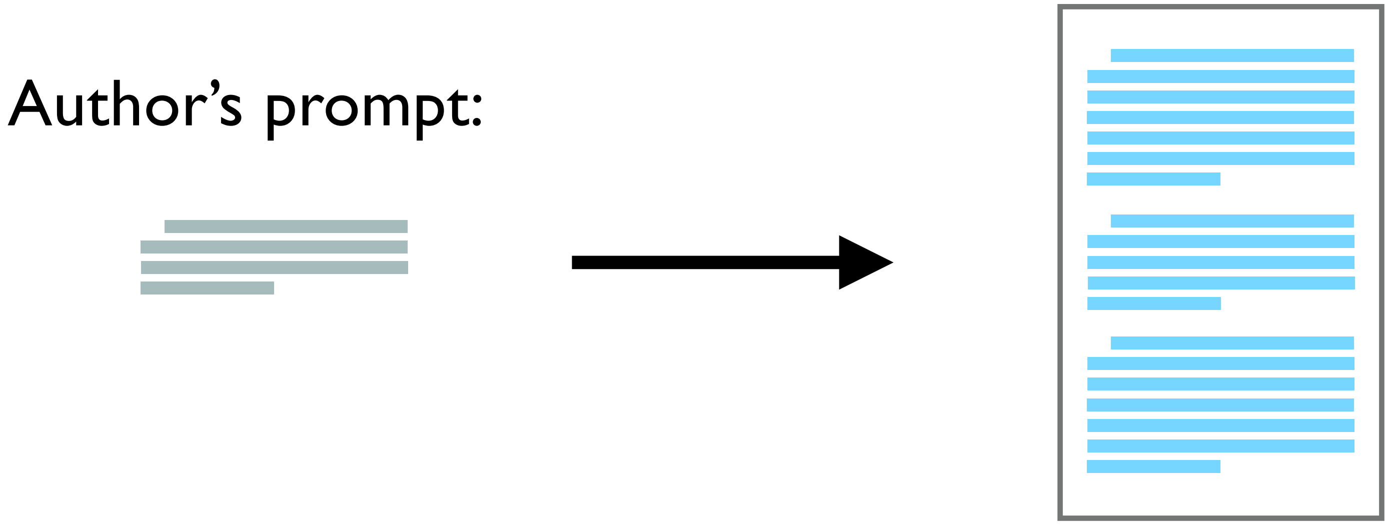
Need #2

Integrate the styles of many different writers

Phantom Pen

Answering the Needs

A program that writes in response to a prompt using the combined styles of many different writers



Author's prompt:

*Created from a compilation of the styles of many writers

How was this built?

(Detailed Version)

Designing Phantom Pen

Phantom Pen

Objective

Build a multifunctional writing program that can not only **generate** new text but can also **classify** user text, **extract** the important sentences and use both of these latter features in generation.

Aspects of the Design

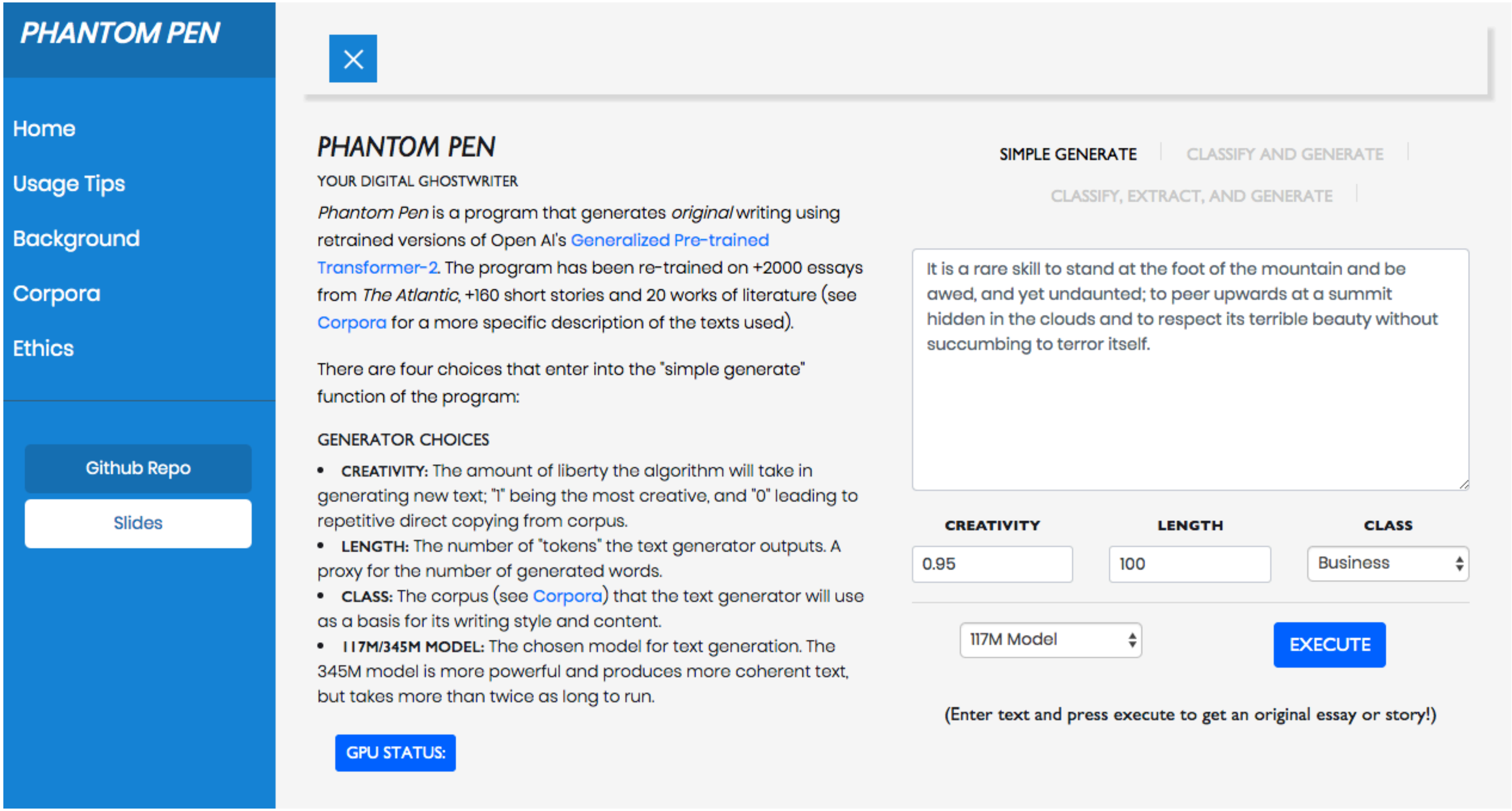
1. Front End
    1. User Interface Structure
    2. Usage Tips
    3. Background
    4. Corpora
    5. Ethics
2. Back End
    1. Language Model
    2. Corpora and Aspects of Training
    3. Classification Model
    4. Extractive Summary

Designing Phantom Pen: Front End

Aspects of the Design

1. Front End
    1. User Interface Structure
    2. Usage Tips
    3. Background
    4. Corpora
    5. Ethics

Creating User Interface



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

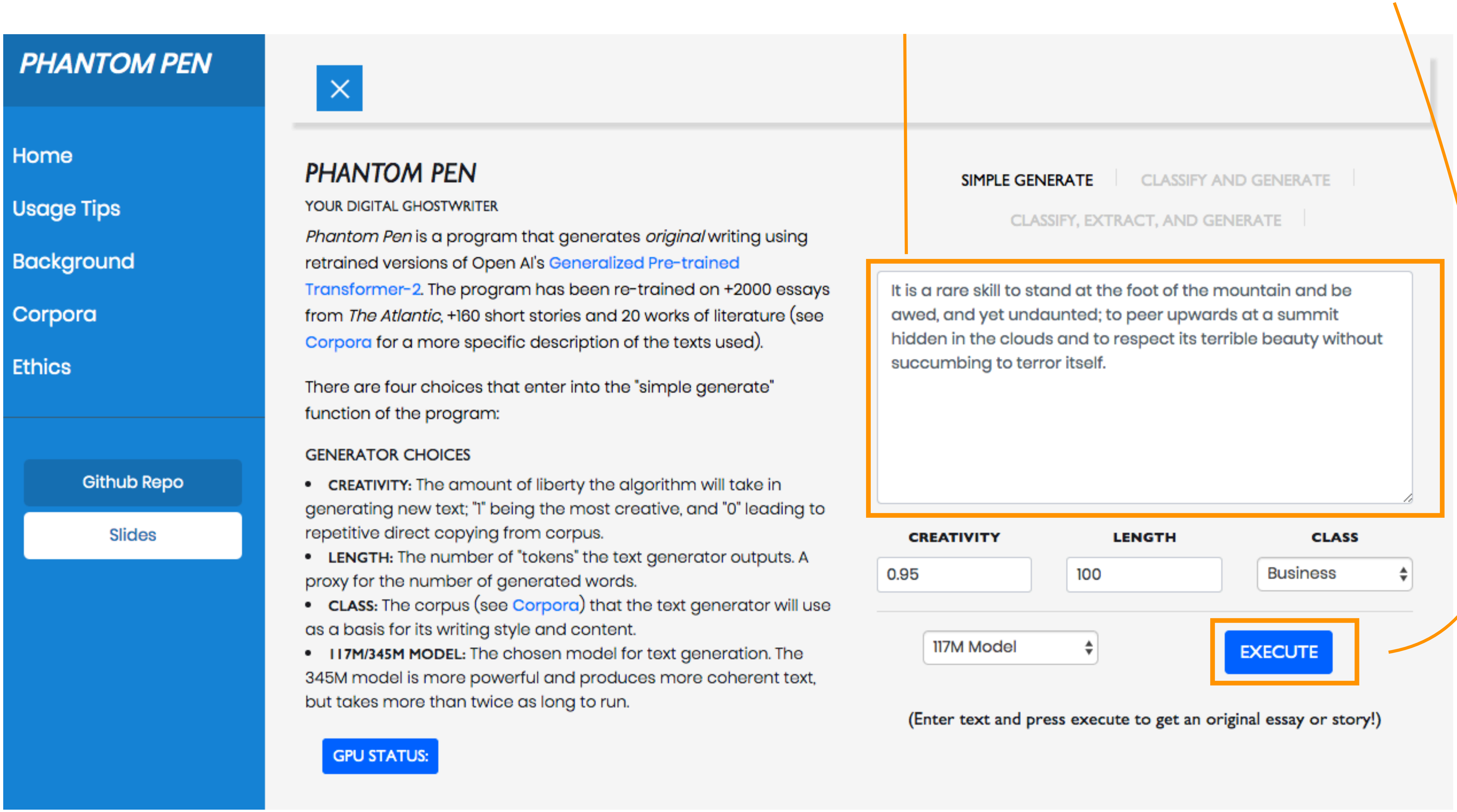
It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY: 0.95
LENGTH: 100
CLASS: Business
117M Model

(Enter text and press execute to get an original essay or story!)

Creating User Interface

It was important to use size and color to emphasize the text-box and highlight the "EXECUTE" button for "impatient" users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY: 0.95

LENGTH: 100

CLASS: Business

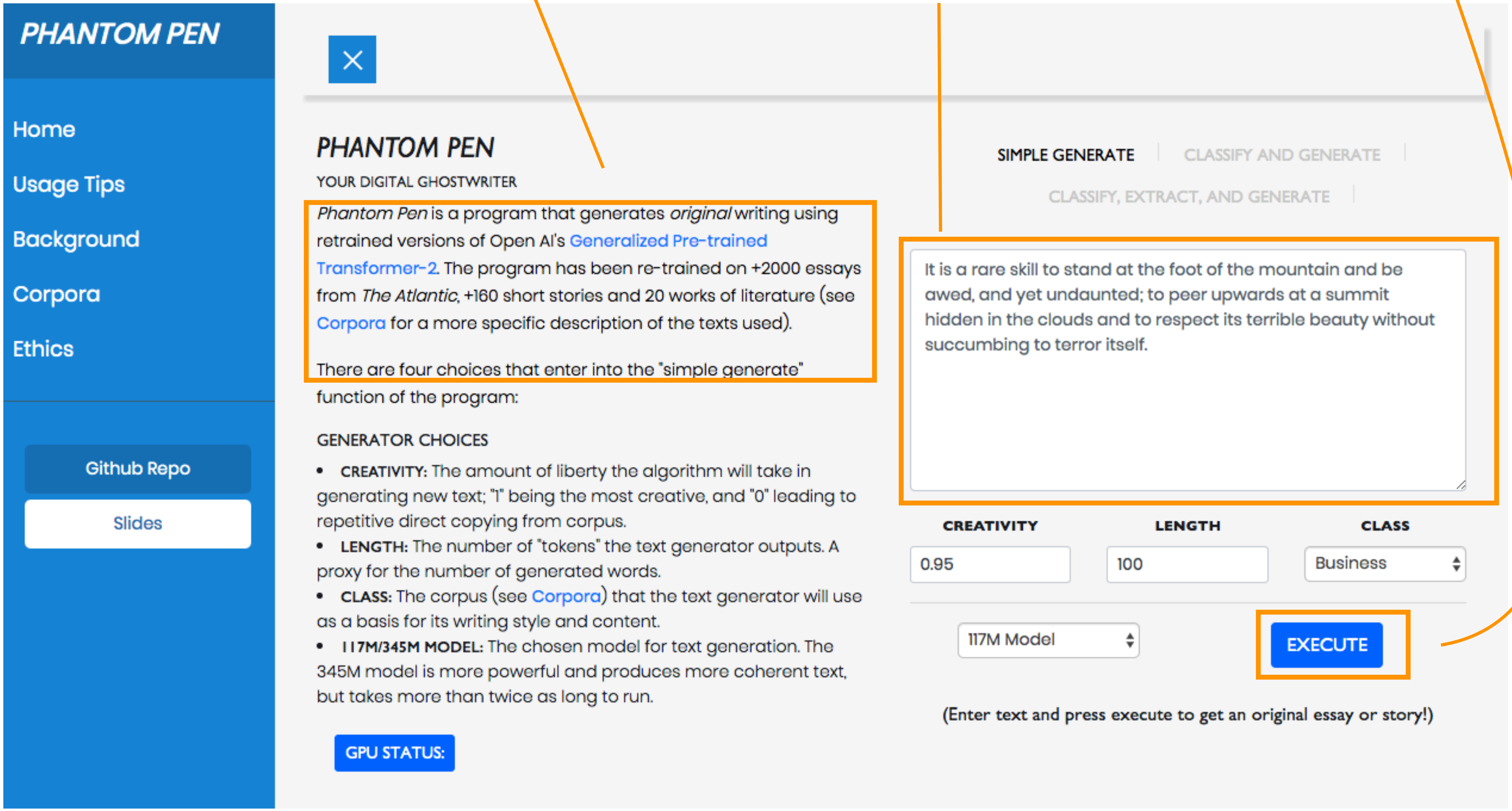
117M Model

(Enter text and press execute to get an original essay or story!)

Creating User Interface

Quick description of what program does

It was important to use size and color to emphasize the text-box and highlight the "EXECUTE" button for "impatient" users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY	LENGTH	CLASS
0.95	100	Business

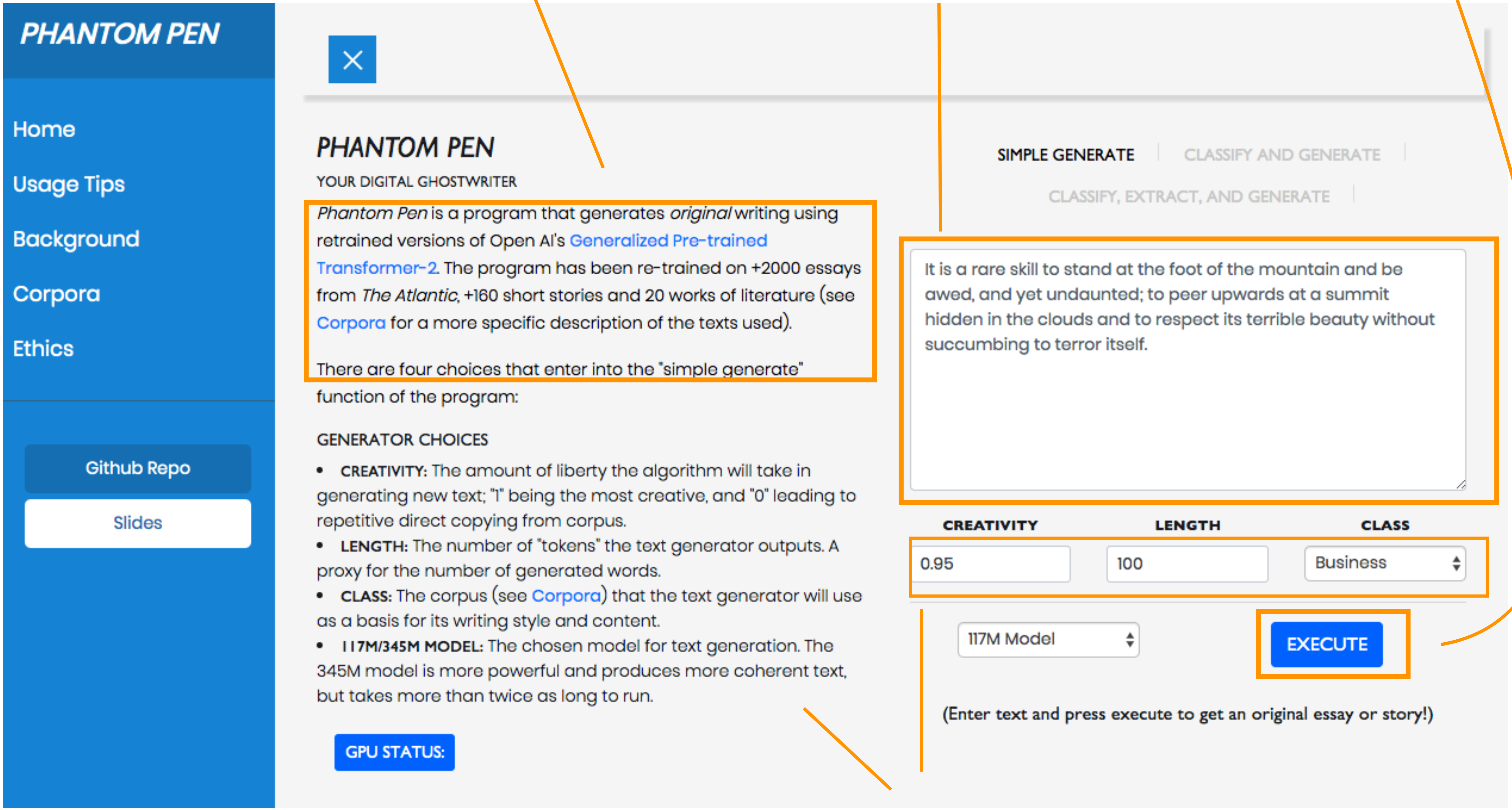
117M Model

(Enter text and press execute to get an original essay or story!)

Creating User Interface

Quick description of what program does

It was important to use size and color to emphasize the text-box and highlight the "EXECUTE" button for "impatient" users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY: 0.95

LENGTH: 100

CLASS: Business

117M Model

(Enter text and press execute to get an original essay or story!)

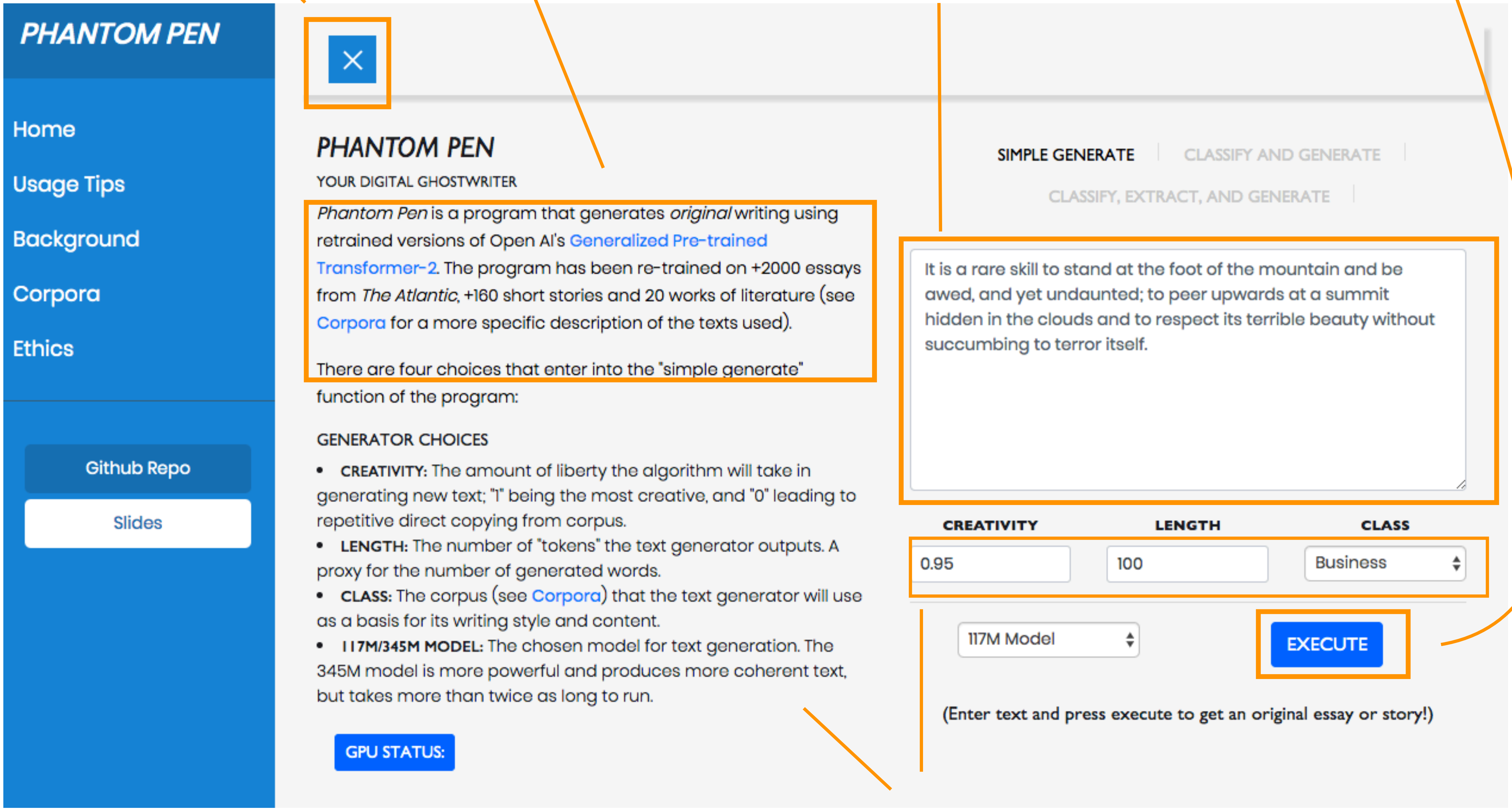
Next to the EXECUTE button are parameter/model choices which are preset. Explanations nearby if interested.

Creating User Interface

Intuitive button to allow for minimizing side-plane

Quick description of what program does

It was important to use size and color to emphasize the text-box and highlight the "EXECUTE" button for "impatient" users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY: 0.95
LENGTH: 100
CLASS: Business

117M Model

(Enter text and press execute to get an original essay or story!)

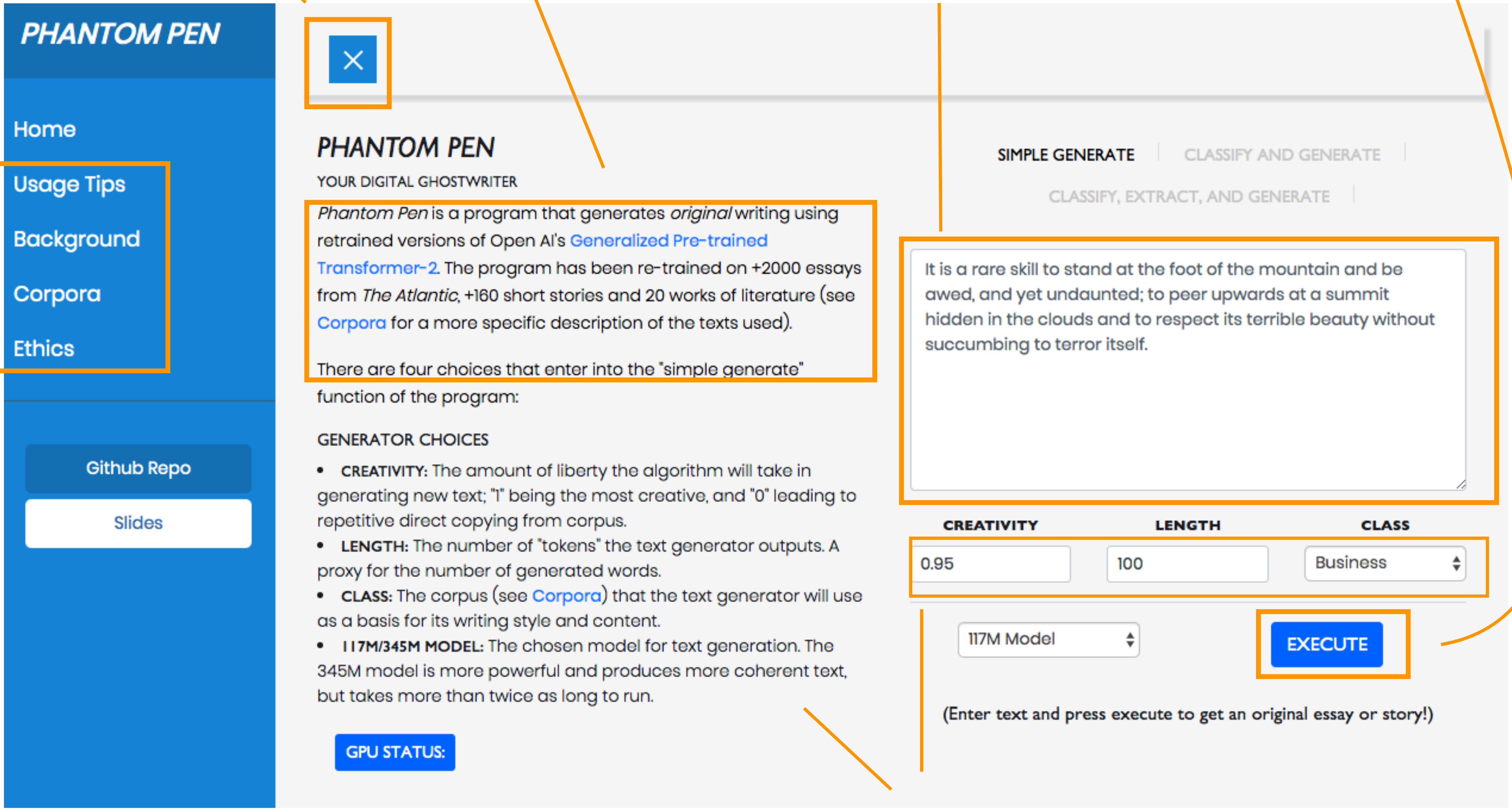
Next to the EXECUTE button are parameter/model choices which are preset. Explanations nearby if interested.

Creating User Interface

Intuitive button to allow for minimizing side-plane

Quick description of what program does

It was important to use size and color to emphasize the text-box and highlight the “EXECUTE” button for “impatient” users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY	LENGTH	CLASS
0.95	100	Business

117M Model

(Enter text and press execute to get an original essay or story!)

The usage tips and background info are out of the way on a different page

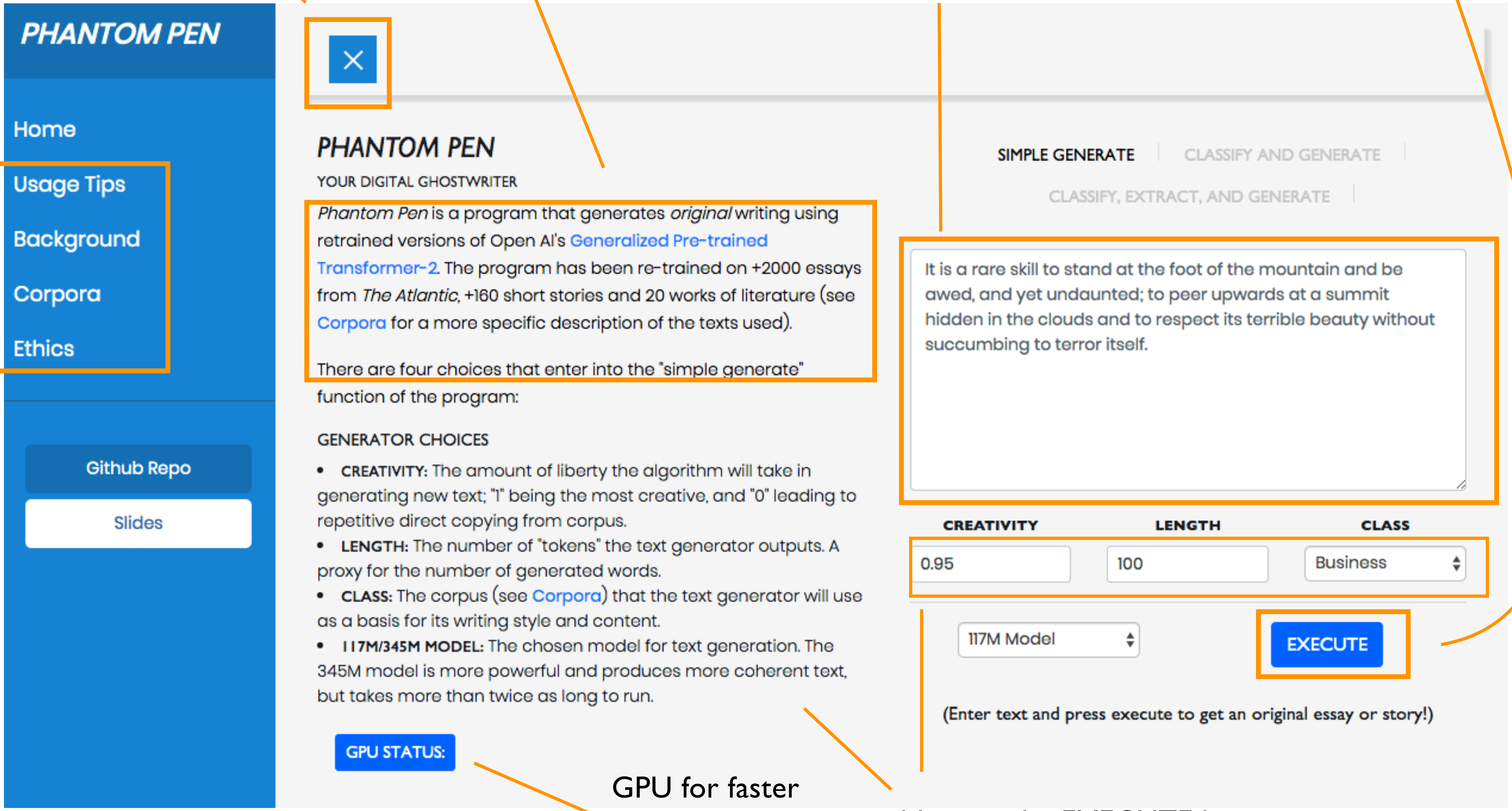
Next to the EXECUTE button are parameter/model choices which are preset. Explanations nearby if interested.

Creating User Interface

Intuitive button to allow for minimizing side-plane

Quick description of what program does

It was important to use size and color to emphasize the text-box and highlight the “EXECUTE” button for “impatient” users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY	LENGTH	CLASS
0.95	100	Business

117M Model

(Enter text and press execute to get an original essay or story!)

GPU for faster text-generation

The usage tips and background info are out of the way on a different page

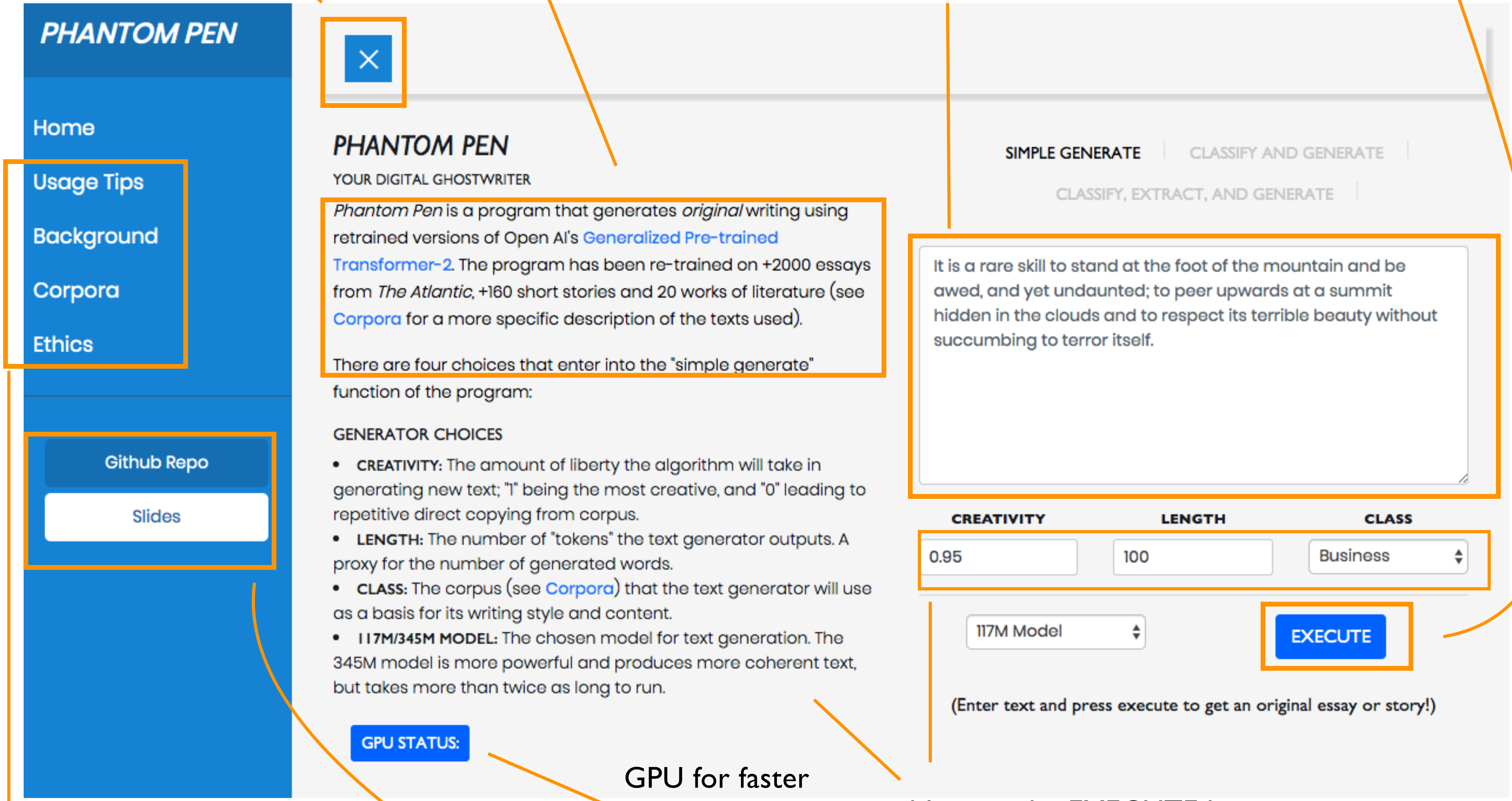
Next to the EXECUTE button are parameter/model choices which are preset. Explanations nearby if interested.

Creating User Interface

Intuitive button to allow for minimizing side-plane

Quick description of what program does

It was important to use size and color to emphasize the text-box and highlight the "EXECUTE" button for "impatient" users



PHANTOM PEN

YOUR DIGITAL GHOSTWRITER

Phantom Pen is a program that generates *original* writing using retrained versions of Open AI's Generalized Pre-trained Transformer-2. The program has been re-trained on +2000 essays from *The Atlantic*, +160 short stories and 20 works of literature (see Corpora for a more specific description of the texts used).

There are four choices that enter into the "simple generate" function of the program:

GENERATOR CHOICES

- **CREATIVITY:** The amount of liberty the algorithm will take in generating new text; "1" being the most creative, and "0" leading to repetitive direct copying from corpus.
- **LENGTH:** The number of "tokens" the text generator outputs. A proxy for the number of generated words.
- **CLASS:** The corpus (see Corpora) that the text generator will use as a basis for its writing style and content.
- **117M/345M MODEL:** The chosen model for text generation. The 345M model is more powerful and produces more coherent text, but takes more than twice as long to run.

It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

CREATIVITY	LENGTH	CLASS
0.95	100	Business

117M Model

(Enter text and press execute to get an original essay or story!)

GPU for faster text-generation

The usage tips and background info are out of the way on a different page

For developers

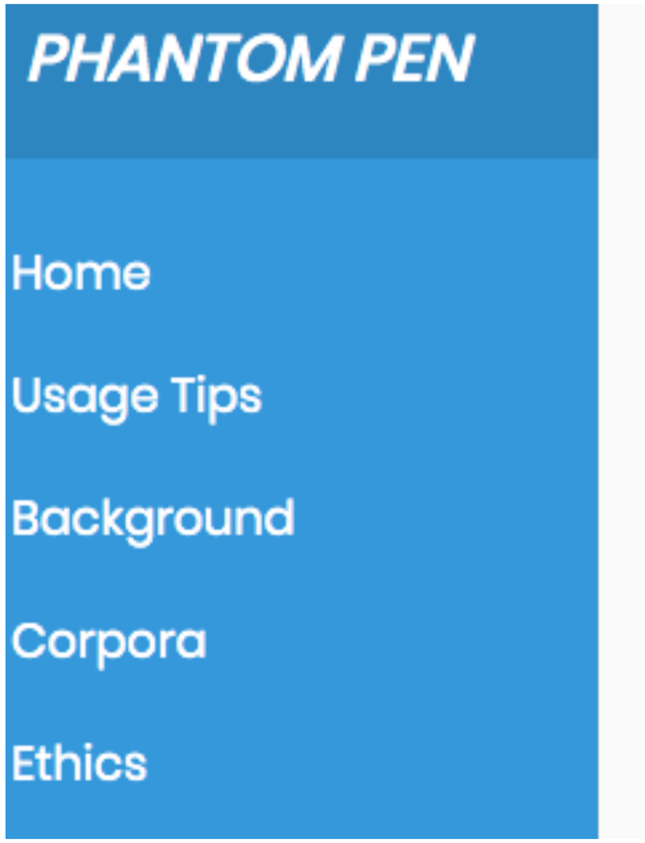
Next to the EXECUTE button are parameter/model choices which are preset. Explanations nearby if interested.

Designing Phantom Pen: Front End

Aspects of the Design

1. Front End
    1. User Interface Structure
    2. Usage Tips
    3. Background
    4. Corpora
    5. Ethics

Motivation for Including Additional Sections



Usage Tips:

From my own experience creating and testing the application, I learned useful generation strategies that would help a fresh-user (and would have helped me) obtain better outputs from the program.

Background:

Most people have seen text generators before, and most of these generators are unimpressive. Thus as background I thought it would be useful to explain how the algorithm at the background of this application is different from the others

Corpora:

The most important section from a text generation perspective. The output of the generator is highly dependent on the corpora it has been trained on so the user should be able to know what exactly constitutes that corpora

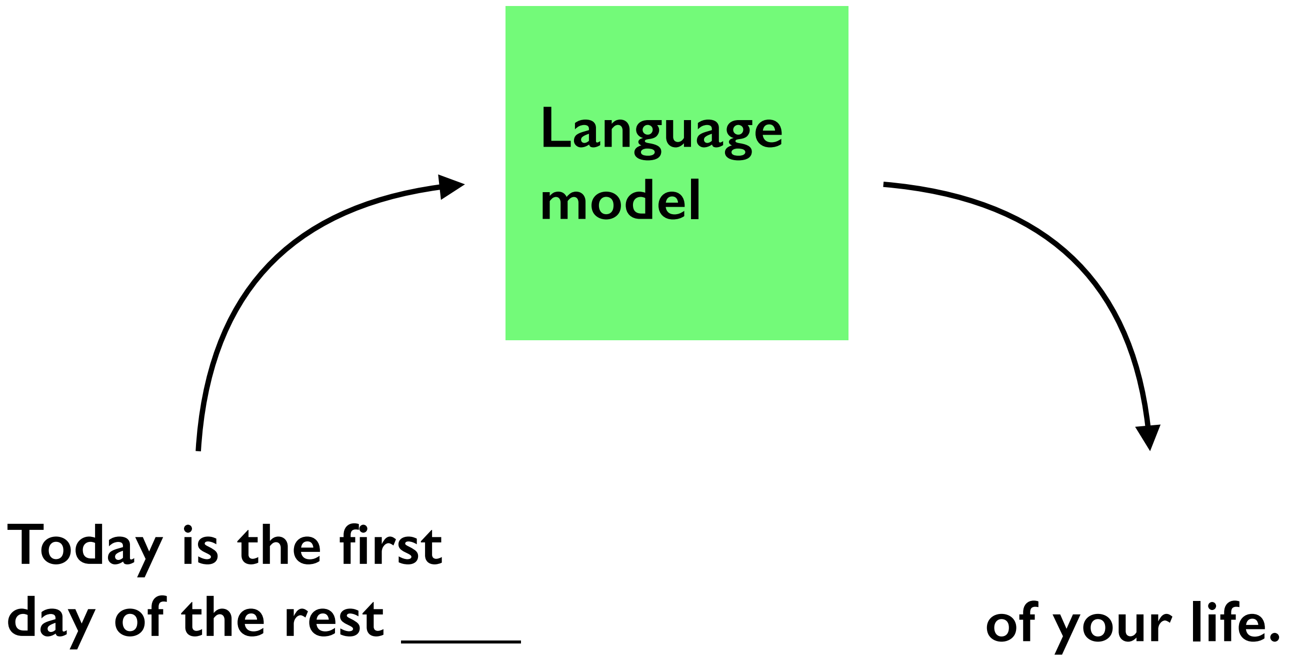
Ethics:

Besides "How do you validate this?" The most common question I received about this program is "How would you deal with plagiarism?" So I wanted to discuss this ethical question and some larger ones surrounding the application.

Designing Phantom Pen: Back End

Aspects of the Design

2. Back End
   1. Language Model
   2. Corpora and Aspects of Training
   3. Classification Model
   4. Extractive Summary



Today is the first day of the rest ____

Language model

of your life.

Language Model - Choices

Possibilities (besides Transformer Model)

1. Markov Chain Monte Carlo
2. Recurrent Neural Network

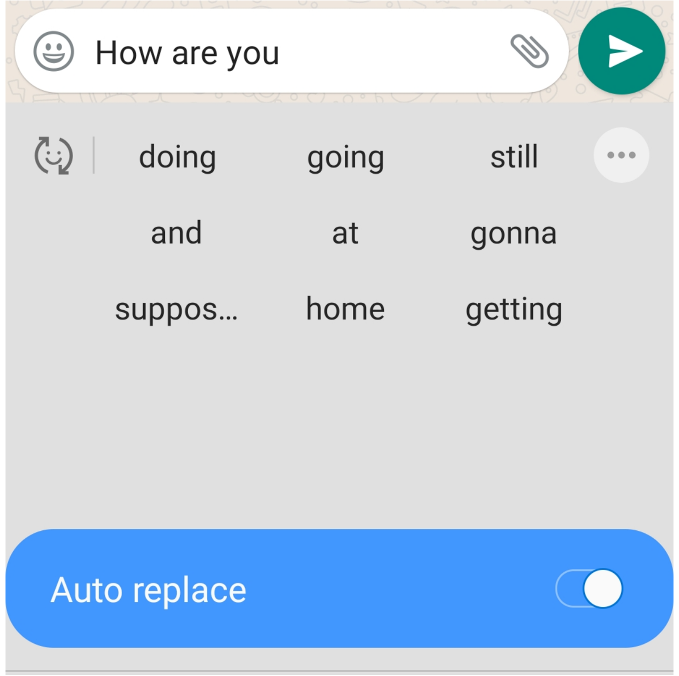
Markov model cannot model context

Possibilities (besides Transformer Model)

1. Markov Chain Monte Carlo
2. Recurrent Neural Network

Comments about Model

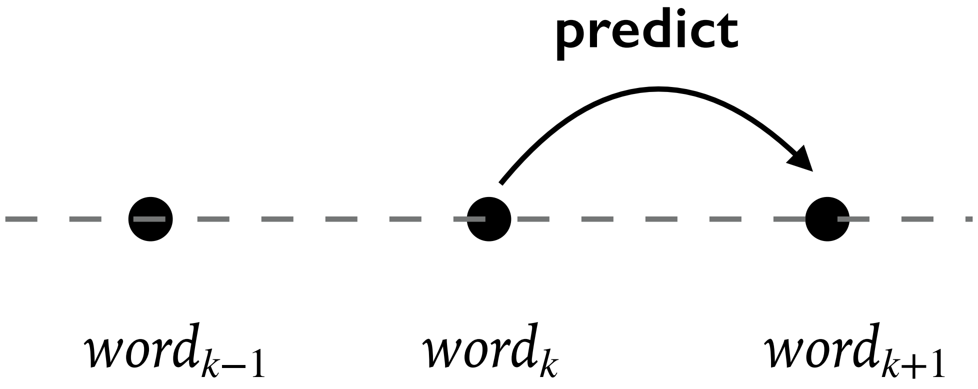
- Conceptually the simplest model to explain
- But, because of its context independence it clearly does not work for generating human-like text.
- Instead, it was used as a baseline model to practice text generation



How are you

doing	going	still
and	at	gonna
suppos...	home	getting

(WhatsApp)

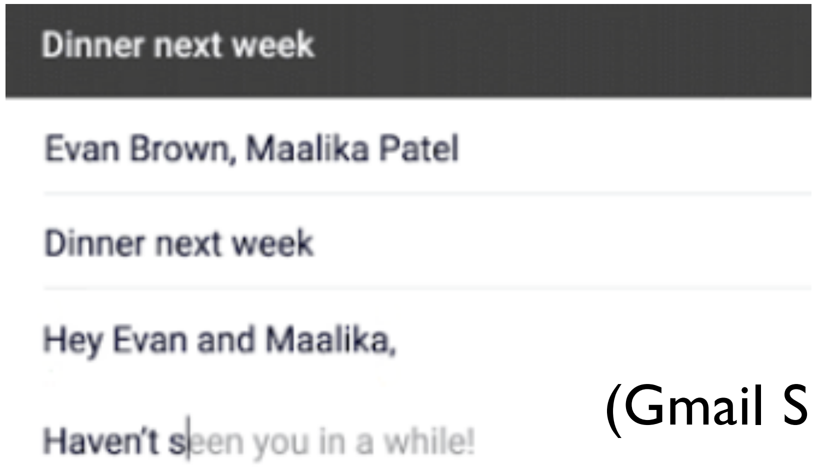


Markov Chain Monte Carlo

Recurrent Neural Network: Better but not excellent

Possibilities (besides Transformer Model)

1. Markov Chain Monte Carlo
2. Recurrent Neural Network



Dinner next week

Evan Brown, Maalika Patel

Dinner next week

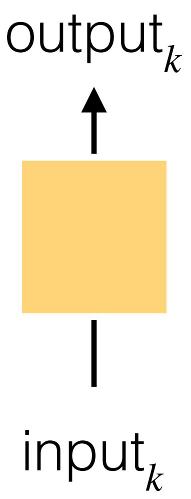
Hey Evan and Maalika,

Haven't seen you in a while!

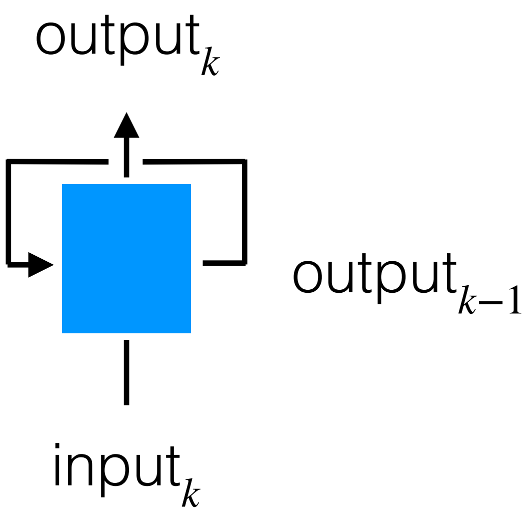
(Gmail Smart Compose)

Comments about Model

- With the Long Short-Term Memory (LSTM) node architecture, RNN's were the SOTA in language model generation
- Clearly much better than Markov Model but still doesn't "sound" like something a human would write (even with LSTM)
- Was the model I was going to use, before I discovered the transformer model



Vanilla Neural Network



Recurrent Neural Network

Output after Training on *Atlantic: Education* Corpora

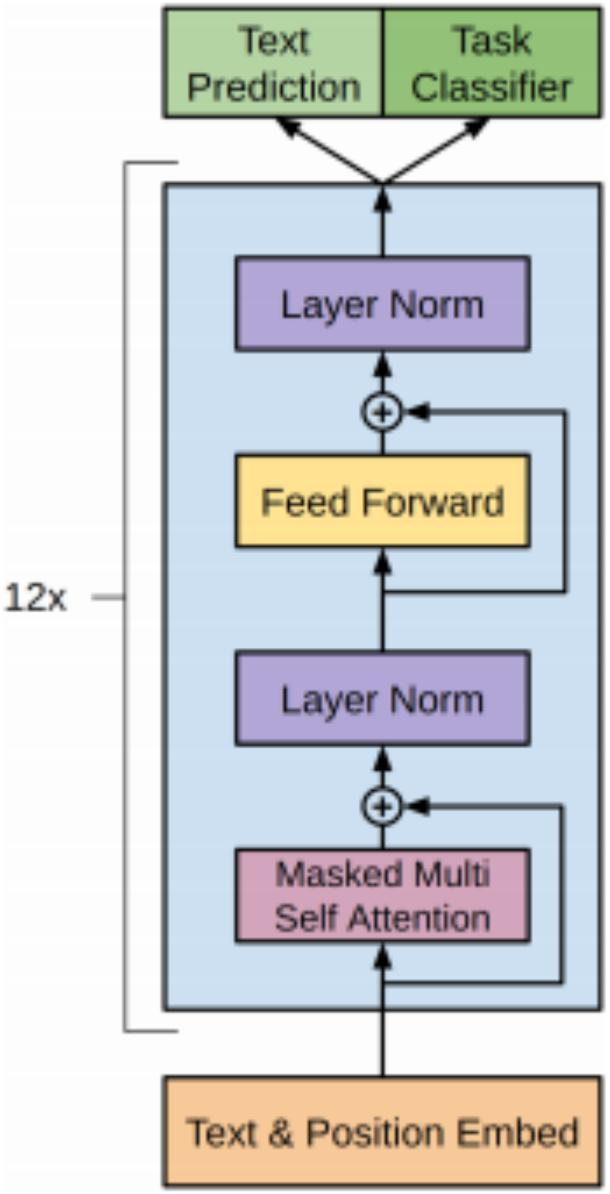
```
As they develop admissions
discussion about health on the
college in the four-year college
are a lettor who were shown about
an extremely policy off the feeling
about their present.
```

Transformer Model

Transformer Model

Comments about Model

- Introduced by Google Brain in 2017, and quickly thereafter became SOTA
- **"Attention"** is the key feature of the model
- Open AI released, "small" and "medium" parameter versions of their Transformer Language Model Called Generalized Pre Trained Transformer (GPT)
- Does not propagate hidden states from input to input as in an RNN



Generalized Pre-Trained Transformer (GPT) Architecture

Network Architecture

117M Hparams (Small)

- 12 layers
- 12 Attention Heads
- 768 dimensional embedding
- 1024 length context vector (for position embedding)
- 50257 vocabulary size

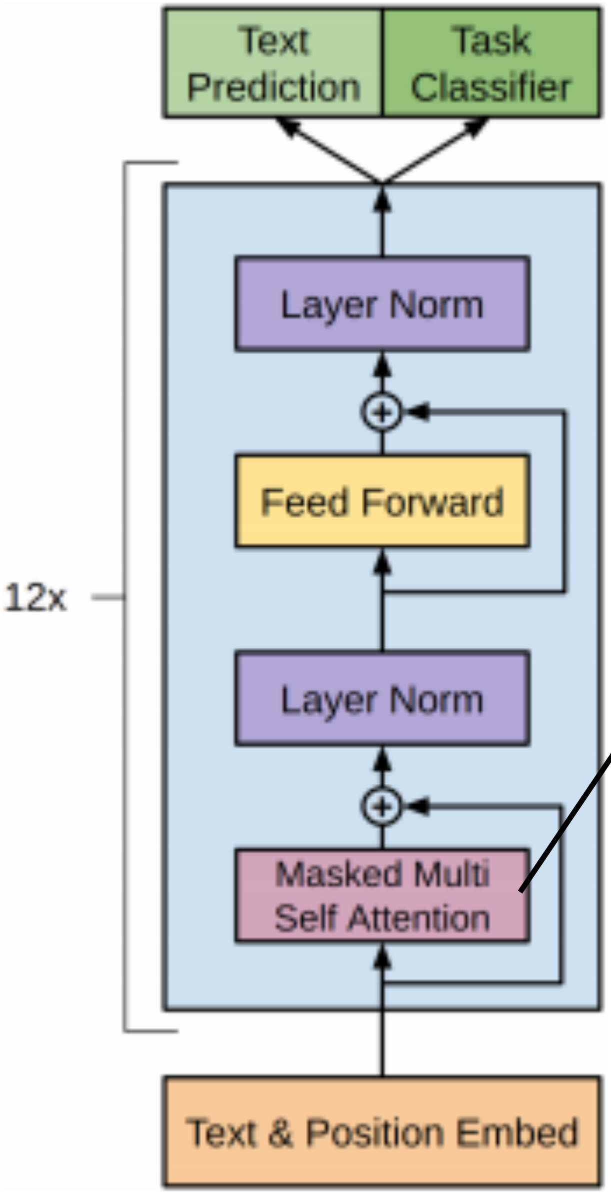
345M Hparams (Medium)

- 24 layers
- 16 Attention Heads
- 1024 dimensional embedding
- 1024 length context vector (for position embedding)
- 50257 vocabulary size (for word embedding)

Transformer Model - Multi headed Attention

Transformer Model

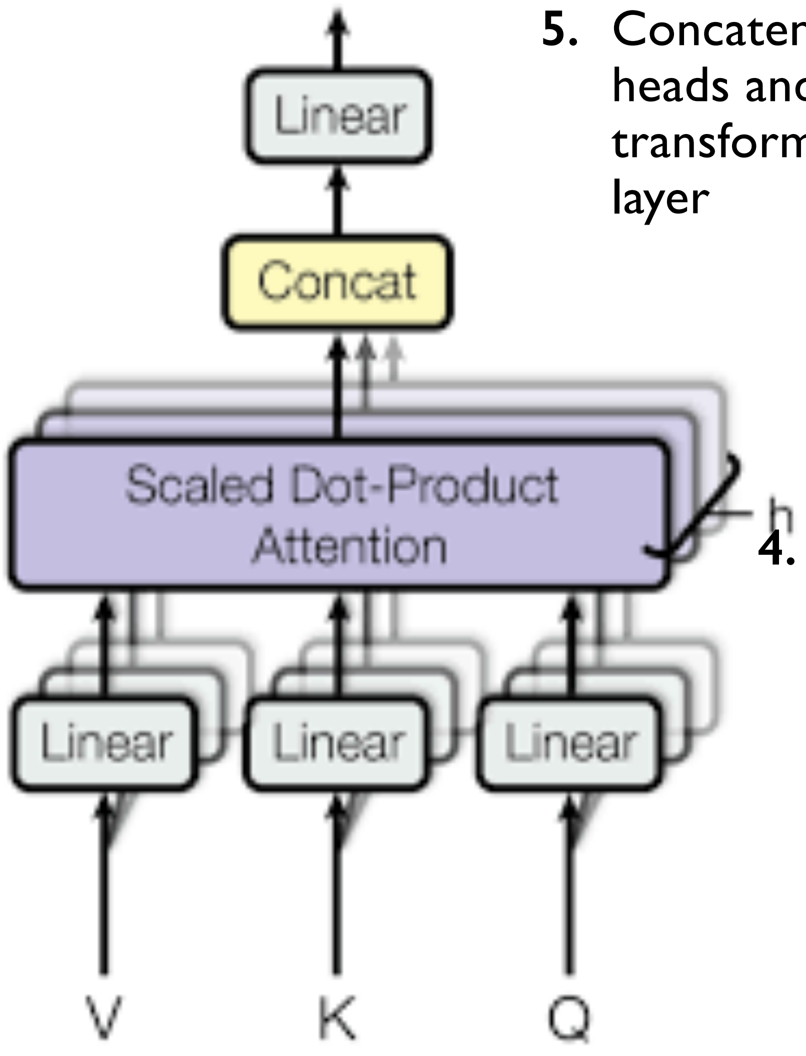
- Multi headed attention



$$softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

3. Softmax calculation to create output of head

2. V, K, and Q matrices are various matrix transforms of the embeddings



5. Concatenate results of heads and linearly transform for next NN layer

4. Perform these operations in parallel on multiple heads

1. Position + Word Embedding

Designing Phantom Pen: Back End

Aspects of the Design

2. Back End
    1. Language Model
    2. Corpora and Aspects of Training
    3. Classification Model
    4. Extractive Summary

2000+ Articles

160 Short Stories

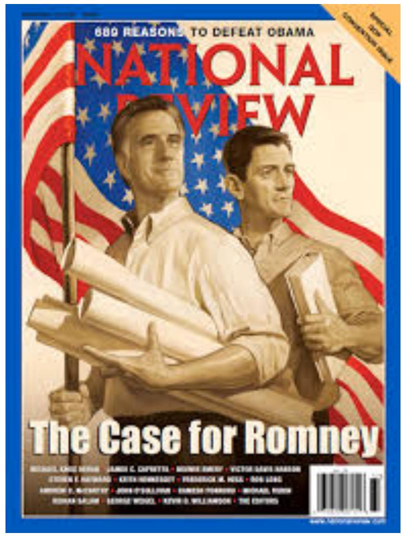
20 Books (Gutenberg)

Corpora and Aspects of Training

The initial plan was to use writing from *SLATE, The Atlantic, National Review, SLATE,* and *NY POST* as the various corpora

Slate

The Atlantic



689 REASONS TO DEFEAT OBAMA

NATIONAL REVIEW

The Case for Romney

NEW YORK POST

Just a Buck!

LATE CITY FINAL

Subway Jared underage sex shock

ENJOY A FOOT LONG IN JAIL

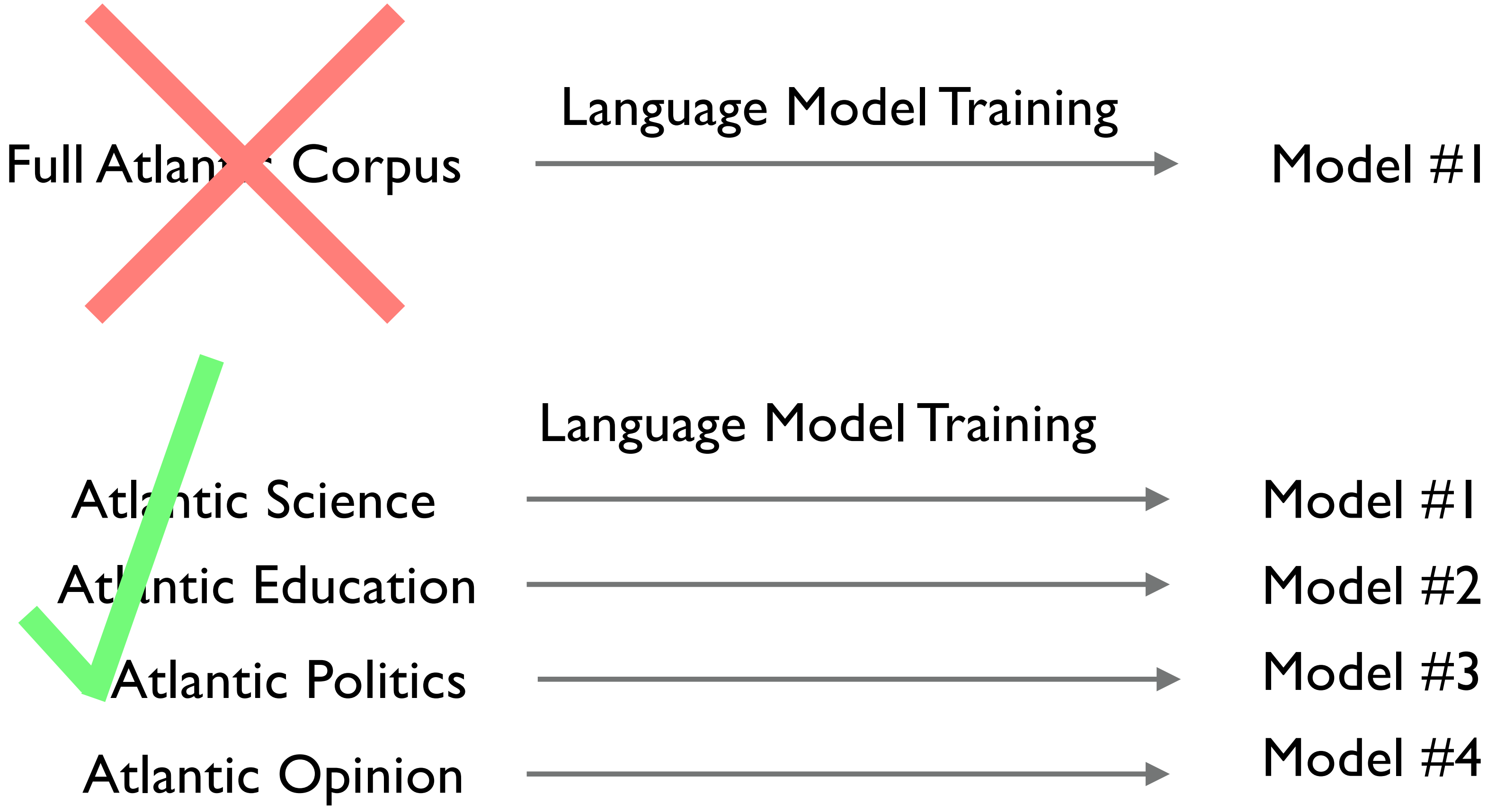
Problems with selecting these corpora

1. There wasn't a sufficiently good use case to motivate these specific corpora
2. The diversity in topics is too much in the direction of the original GPT2 data set of 40 GB of internet text (i.e., no real advantage in re-training)

Corpora and Aspects of Training

I later decided it would make more sense to select corpora according to **topic** rather than publication

This way the application's essay completions/generations are **more specific to a user chosen class** and hence more useful to the user



Full Atlantic Corpus

Language Model Training

Model #1

Language Model Training

Atlantic Science → Model #1

Atlantic Education → Model #2

Atlantic Politics → Model #3

Atlantic Opinion → Model #4

*Sacrificed ideological diversity for output specificity

Improvements: Find a way to train a single model, rather than 2 X 11 different models.

Corpora and Aspects of Training

Introduced **short story** and **Project Gutenberg** corpora and trained models to give the user more options.

2000+ Articles

Nine Article Categories from *The Atlantic*

- Business
- Education
- Entertainment
- Health
- Ideas
- International
- Politics
- Science
- Technology

160 Short Stories

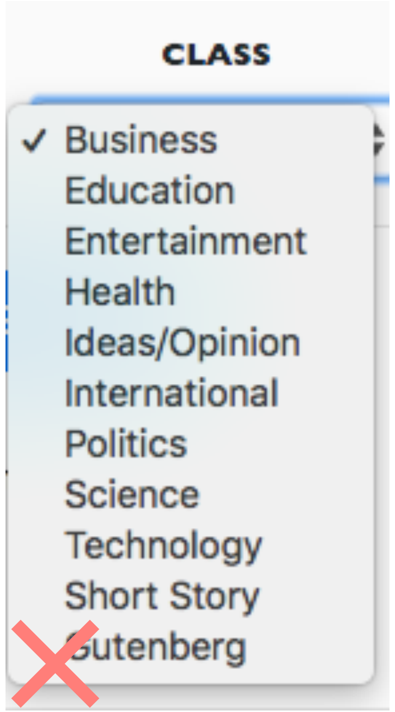
62 Authors

- Anton Chekov
- Henry David Thoreau
- Jack London
- H.P. Lovecraft

20 Books (Gutenberg)

Including

- Pride and Prejudice
- The Count of Monte Cristo
- Robinson Crusoe
- Moby Dick

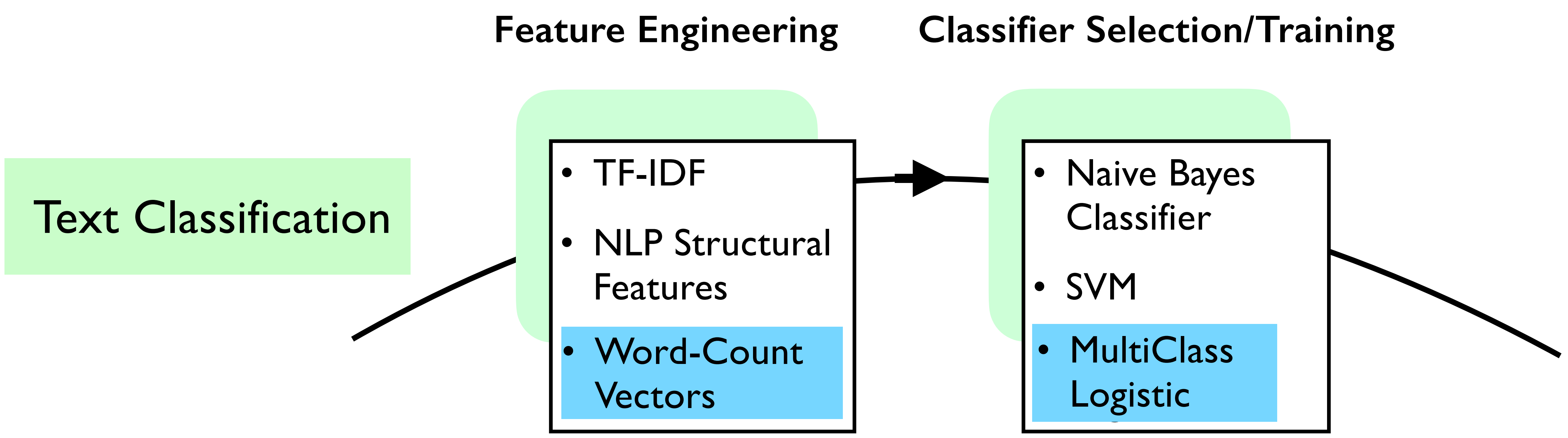


Improvements: In a second version of the model, I would **remove the Gutenberg class** since its texts are structurally different from the other texts

Designing Phantom Pen: Back End

Aspects of the Design

2. Back End
    1. Language Model
    2. Corpora and Aspects of Training
    3. Classification Model
    4. Extractive Summary



Text Classification

Feature Engineering

- TF-IDF
- NLP Structural Features
- Word-Count Vectors

Classifier Selection/Training

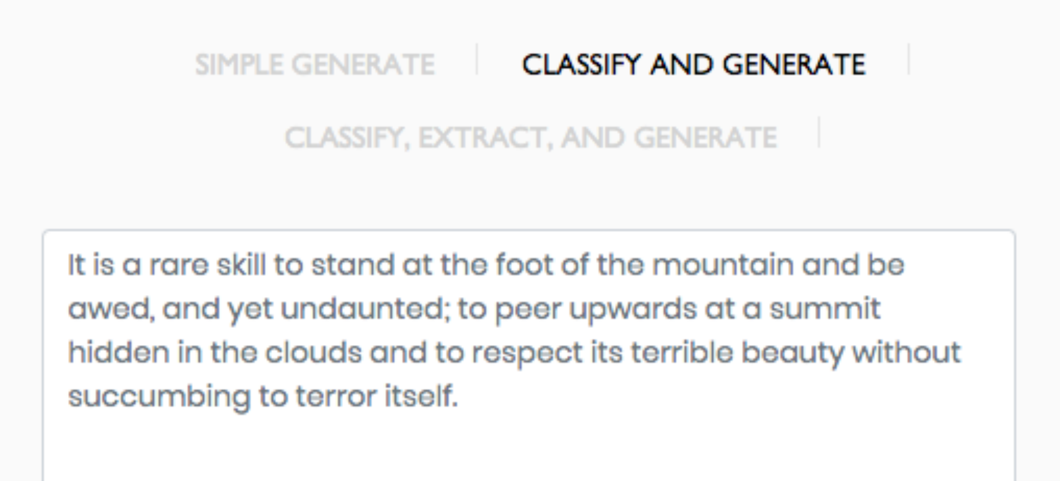
- Naive Bayes Classifier
- SVM
- MultiClass Logistic

Classifying Text for the User

Beyond simple text generation, I wanted to add additional functions to allow for more flexible user input

Additional Functionality #2:
Determine the class for the user

Program screenshot



It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

Text Classification Problem

We want to determine which of the 10 classes the user's entered text is most similar to

Feature Engineering	**Classifier Model**
Word Count Vectors	Naive Bayesian Classifier
Word-Embedding	Support Vector Machine
TF-IDF	Logistic Regression

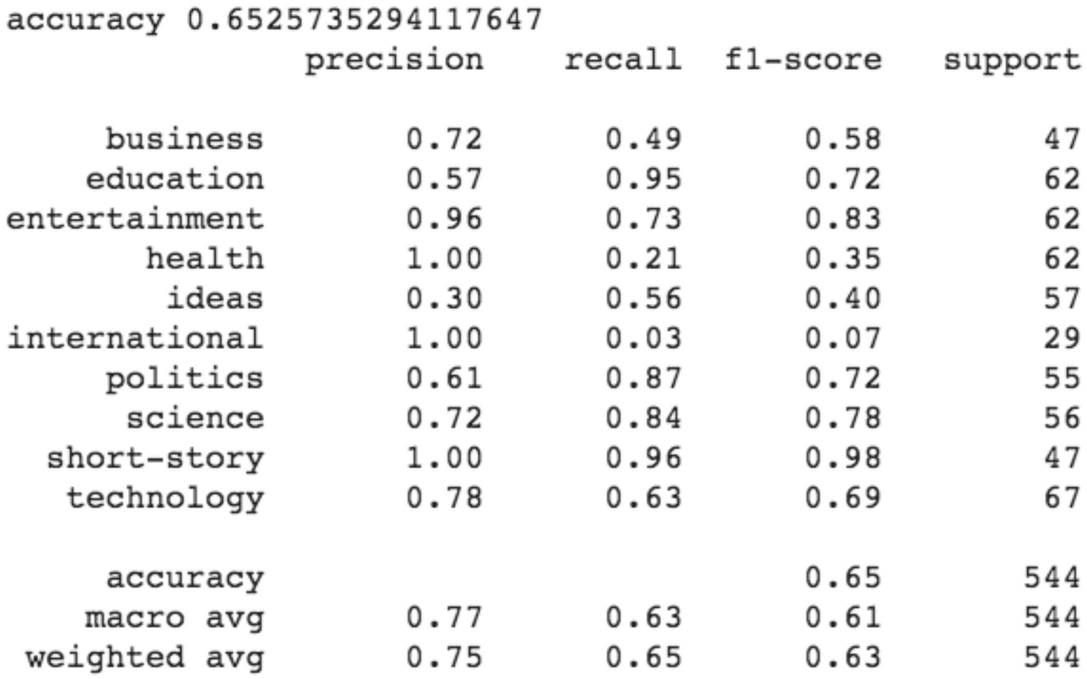
Classifying Text For the User

Feature Engineering	**Classifier Model**
Word Count Vectors	Naive Bayesian Classifier
Word-Embeddings	Support Vector Machine
TF-IDF	Logistic Regression

- Simplest model had the highest accuracy

https://github.com/mowillia/phantom_pen/blob/master/classification_comparisons.ipynb

SVM; Acc = 0.6525

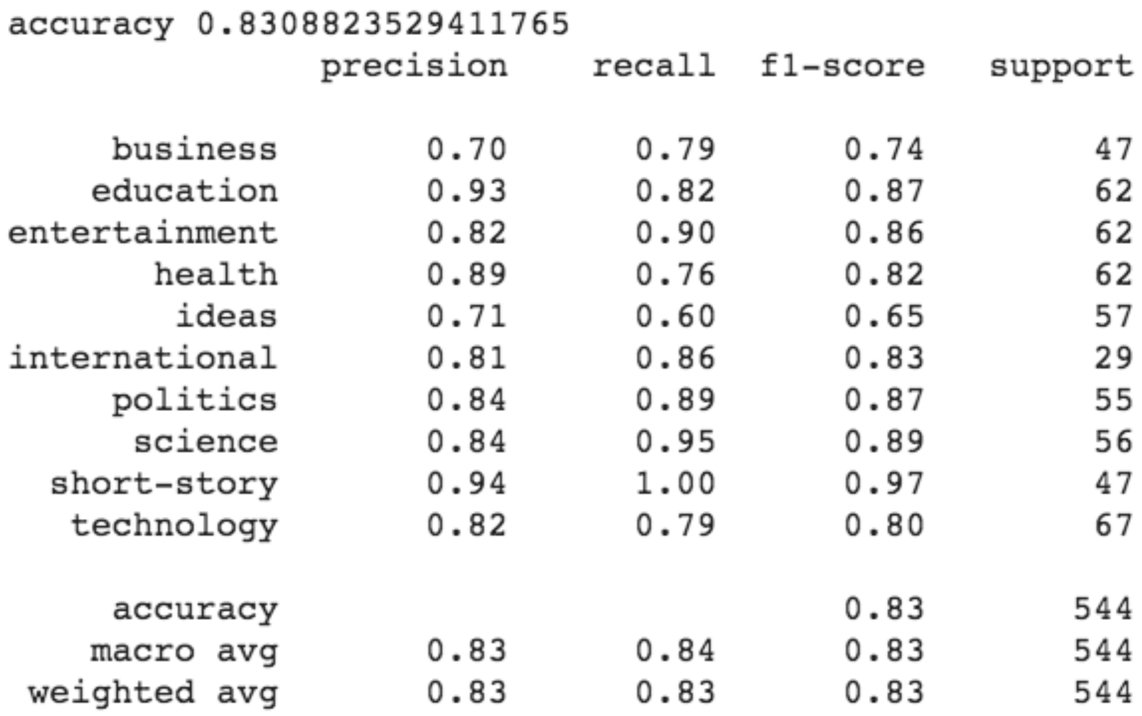


```
accuracy 0.6525735294117647
               precision    recall  f1-score   support

     business       0.72      0.49      0.58        47
    education       0.57      0.95      0.72        62
entertainment       0.96      0.73      0.83        62
       health       1.00      0.21      0.35        62
        ideas       0.30      0.56      0.40        57
international       1.00      0.03      0.07        29
     politics       0.61      0.87      0.72        55
      science       0.72      0.84      0.78        56
  short-story       1.00      0.96      0.98        47
   technology       0.78      0.63      0.69        67

     accuracy                           0.65       544
    macro avg       0.77      0.63      0.61       544
 weighted avg       0.75      0.65      0.63       544
```

Logistic Regression; Acc = 0.831

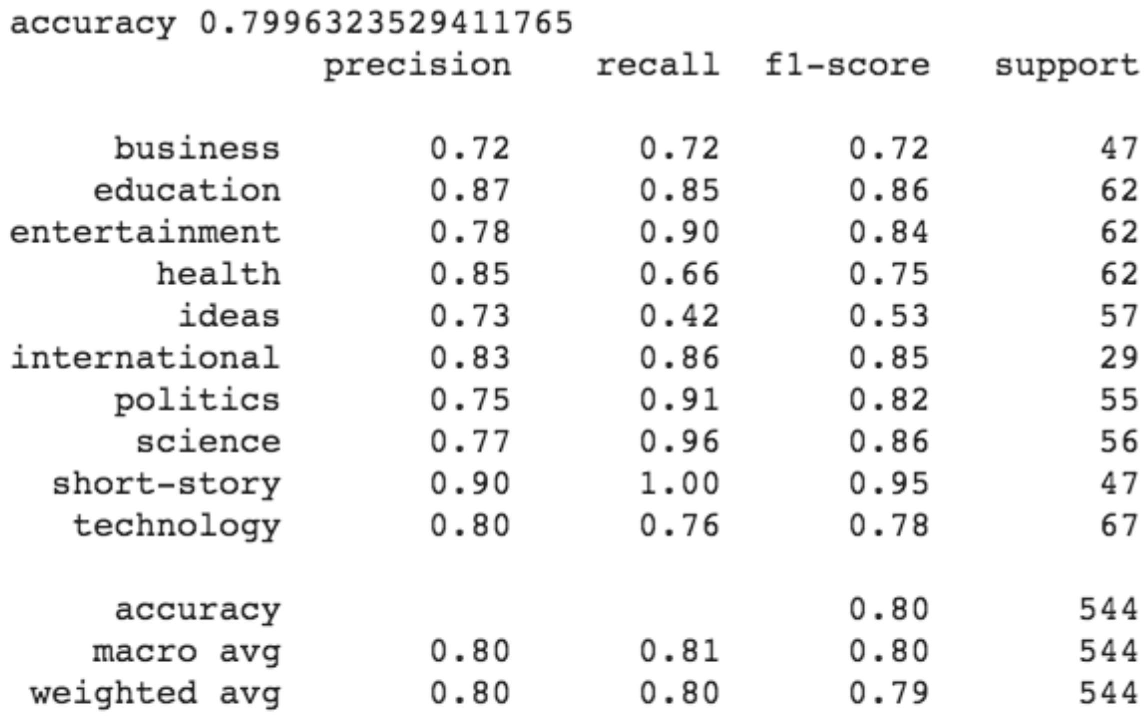


```
accuracy 0.8308823529411765
               precision    recall  f1-score   support

     business       0.70      0.79      0.74        47
    education       0.93      0.82      0.87        62
entertainment       0.82      0.90      0.86        62
       health       0.89      0.76      0.82        62
        ideas       0.71      0.60      0.65        57
international       0.81      0.86      0.83        29
     politics       0.84      0.89      0.87        55
      science       0.84      0.95      0.89        56
  short-story       0.94      1.00      0.97        47
   technology       0.82      0.79      0.80        67

     accuracy                           0.83       544
    macro avg       0.83      0.84      0.83       544
 weighted avg       0.83      0.83      0.83       544
```

Naive Bayes Classifier; Acc = 0.80



```
accuracy 0.7996323529411765
               precision    recall  f1-score   support

     business       0.72      0.72      0.72        47
    education       0.87      0.85      0.86        62
entertainment       0.78      0.90      0.84        62
       health       0.85      0.66      0.75        62
        ideas       0.73      0.42      0.53        57
international       0.83      0.86      0.85        29
     politics       0.75      0.91      0.82        55
      science       0.77      0.96      0.86        56
  short-story       0.90      1.00      0.95        47
   technology       0.80      0.76      0.78        67

     accuracy                           0.80       544
    macro avg       0.80      0.81      0.80       544
 weighted avg       0.80      0.80      0.79       544
```

Designing Phantom Pen: Back End

Aspects of the Design

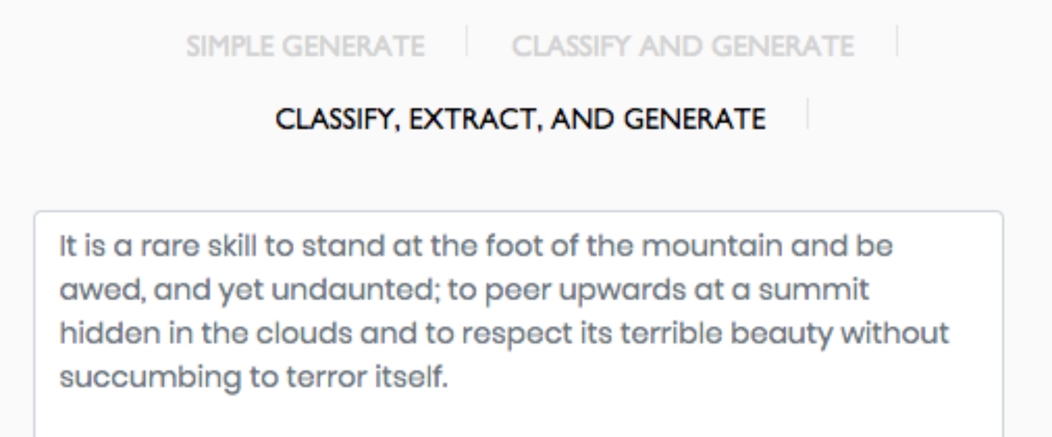
2. Back End
   1. Language Model
   2. Corpora and Aspects of Training
   3. Classification Model
   4. Extractive Summary

Extracting Important Sentences from Text

The program performed less well on longer texts, so I wanted a way to condense longer texts into shorter text

Additional Functionality #3:
Extract important sentences, classify entire text, and generate new text from extraction

Program screenshot



It is a rare skill to stand at the foot of the mountain and be awed, and yet undaunted; to peer upwards at a summit hidden in the clouds and to respect its terrible beauty without succumbing to terror itself.

https://github.com/mowillia/phantom_pen/tree/master

https://github.com/mowillia/phantom_pen/blob/master/text_generation_function.ipynb

Text Extraction Problem

We want to condense a long essay into a shorter one while keeping as much as the important information as possible.

Extractive Summarization

Pick out the sentences that are most important

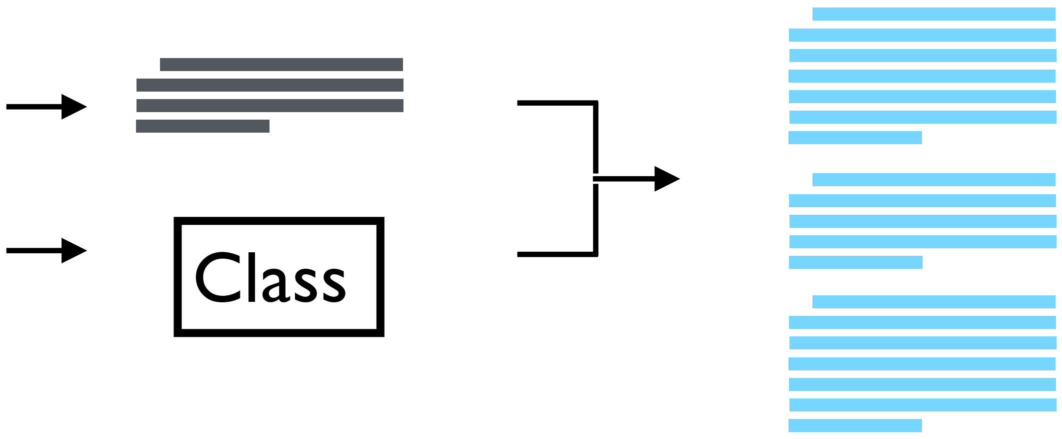
I considered three approaches

- Word Frequency
- Word Embeddings and Cosine Similarity
- Word-count vectors and Cosine Similarity

*Metric of performance based on how well it picks out sentences I considered important

— More rigorous approach: Hand labeling and then checking accuracy

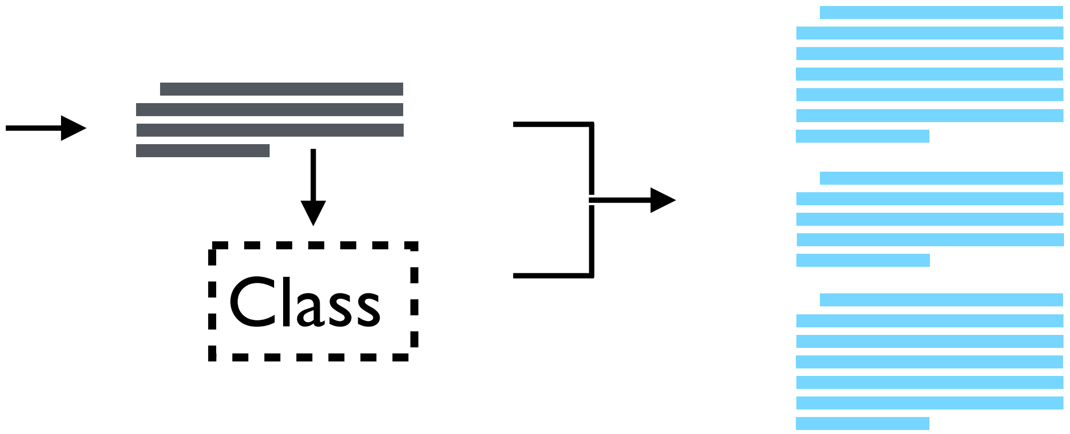
Functions of application



MAIN FUNCTION

SIMPLE GENERATE

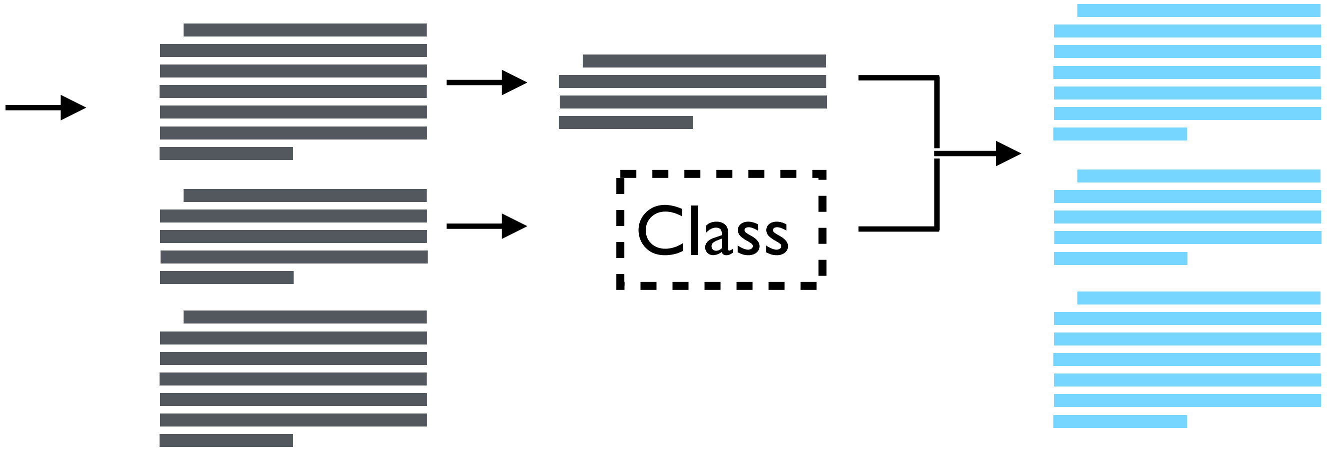
Generates an essay given a **prompt** and a chosen **class**



ADDITIONAL FUNCTION #1

CLASSIFY AND GENERATE

Generates an essay given a **prompt** and the class determined from the prompt



ADDITIONAL FUNCTION #2

CLASSIFY, EXTRACT, AND GENERATE

Generates an essay given a **previous essay**

- Classifies and extracts the important sentences and generates essay from both

END