



# Bible as Interface

## Reading Bible with Machines

MICHAEL HEMENWAY, JUSTIN O. BARBER, SHAWN GOODWIN, MICAH SAXTON  
AND TIMOTHX BEAL

Illiff School of Theology Denver  
mphemenway@gmail.com

*Michael Hemenway, Justin O. Barber, Shawn Goodwin, Micah Saxton, and Timothy Beal explore what we can learn by reading the bible with machines. They present a text generation model that produces narrative commentary on biblical passages. The model demonstrates existing tendencies in bible commentary and elicits new questions and insights on the corpus as “interface.”*

This project is a collaborative contribution of the Experimental Humanities Lab at the Illiff School of Theology and ai.illiff, the AI Institute at Illiff. To engage the fullness of our contribution to the workshop, please visit <https://illiff.github.io/theologiesofthedigital/> and the code repository for the project at <https://github.com/illiff/theologiesofthedigital>. The writing included here is merely one slice of the larger project with data, python code, and model weights constituting the majority of our work. That said, we will share some of our reflections on the project here in hopes of fostering further dialog around the workshop.

### 1. Problem Definition

What can we learn from reading bible with machines? To explore this question, our team is building a text generation model that will take a short passage from bible as

input and output a narrative commentary on the passage. We begin with the assumption that machines can be significant partners in reading corpora like bible by learning from existing commentary data and introducing novel reflections on a given passage. These machine generated reflections on bible passages can teach us about the existing tendencies in bible commentary and can introduce new questions and insights on the corpus.

So, our challenge is to build a machine learning based text generation model that contributes to the community of conversation reflecting on bible passages. In order to build this model, we will leverage some strategies already in development at ai.iliff, the AI Institute at Iliff School of Theology, to produce a conversational AI that can participate in online courses to facilitate better learning outcomes for students.

We believe the process of building, testing, and deploying this model can raise interesting and important questions about the theory and practice of bible interpretation in a technological era where machine learning will increasingly participate in our reading and research. Some such questions include:

- Does our theology of scripture change when we partner with machines in reading bible?
- What is the role of historical reflections on bible in our contemporary interpretive traditions and practices? In a sense, how do we use our data in our own interpretive practice?
- What constitutes the boundary between bible and our reading/interpretation of it?
- How does the digital materiality of bible, no longer constrained by the codex, change the ways we use and understand bible?
- How does reading and writing with machines impact our notions of authorship, creativity, and interpretive capacity?
- How might the machine's processes of generating theological commentary (dependent on and derivative of the inherited tradition/corpus of theological discourse on which it trains) invite reflection on our own human processes of theological reflection and the generation of new theological ideas?

For the first iteration of this model, to demonstrate proof of concept, we will build a very simple interface that will allow a user to give the text of a biblical passage to the model and in return, the model will construct a response, word by word, that

comments on the given passage. These machinic readings of bible will hopefully foster conversation among scholars and practitioners that will in turn help us develop more useful iterations of the model.

## 2. Technologies

In the interest of being intentional about the materialities at work in this project and to foster other's reproduction and expansion on this work, we will define the main technologies we will use to build this project.

### 2.1. Python

Python is a highly readable programming language, particularly useful for data science and machine learning.<sup>1</sup> As we have outlined in "Library as Interface for DH Work,"<sup>2</sup> we have invested in Python as a core competency in our Experimental Humanities Lab for several reasons. The most important reasons for choosing Python for this particular project are Python's extensive set of openly available libraries to work with natural language processing tasks such as data preparation and with cutting edge deep learning frameworks such as PyTorch and gpt-2 (see below). The focus on readability in Python also provides an excellent environment for teaching and collaboration, lowering the barriers for people to engage the code.

### 2.2. GPU

One of the major advances in machine learning computation power over the past several years has been the growing accessibility of graphical processing units (GPU). Mythbusters provide a very useful enactment of the difference between a GPU and a more traditional central processing unit (CPU).<sup>3</sup>

The main advantage of a GPU for machine learning tasks is the capability for parallel processing. Structures like neural networks with many layers and millions of parameters require a large amount of processing. Leveraging a GPU allows us to train models

<sup>1</sup> See <https://www.python.org/> (accessed: May 18, 2020).

<sup>2</sup> Experimental Humanities @ Iliff, "Library as Interface for DH Work," in Clifford B. Anderson, ed., *Digital Humanities and Libraries and Archives in Religious Studies* (Berlin: De Gruyter, 2019).

<sup>3</sup> "GPU vs. CPU," <http://www.youtube.com/watch?v=-P28LKWTzrI> (accessed: November 16, 2021).

on large data sets much faster so we can iterate through experiments more quickly to optimize our models.<sup>4</sup>

Thanks to generous funding from the Henry Luce Foundation, Iliff's AI Institute has a dedicated GPU server with an NVidia Tesla V100 GPU<sup>5</sup> and 32GB of memory that we are able to use to train our models for this project.

### 2.3. gpt-2

In the last few years, we have seen an explosion in the development of powerful pre-trained language models that can be used as a foundation for several natural language processing tasks, such as text generation, question answering, machine translation, and more. The two language models we use in our development currently are BERT from Google and gpt-2<sup>6</sup> from OpenAI. These language models provide a statistical representation of a language (we are currently focused on English) that can be tuned to a specific discourse and then used for several tasks.<sup>7</sup>

For this bible commentator project, we are using gpt-2 to generate text one word at a time in response to a prompt verse. Trained on text from 1.5 million web pages to generate a generic language model, gpt-2 prevents us from having to build a language model from scratch on a much smaller dataset. Instead, we can build on top of the generic language model to tune the model toward our particular task by training gpt-2 using our particular bible commentary corpora.

If you would like to see a demonstration of gpt-2 text generation in response to an input, you can visit Adam King's simple web interface for gpt-2.<sup>8</sup> Just enter a prompt and see what the model writes.

---

<sup>4</sup> For more information on why a GPU can be useful in machine learning applications, there are many explanations available online. A good example is Faizan Shaikh's "Why are GPUs necessary for training Deep Learning models?", <https://www.analyticsvidhya.com/blog/2017/05/gpus-necessary-for-deep-learning/> (accessed: November 16, 2021).

<sup>5</sup> See <https://www.hluce.org/programs/theology/>, <https://ai.iliff.edu>, and <https://images.nvidia.com/content/technologies/volta/pdf/tesla-volta-v100-datasheet-letter-fnl-web.pdf> respectively (accessed: November 16, 2021).

<sup>6</sup> See <https://arxiv.org/abs/1810.04805> and <https://openai.com/blog/better-language-models/> (accessed: November 16, 2021).

<sup>7</sup> The vectorization of language used in the embedding techniques of these language models deserves far more attention than we can give it here. The ways machines can represent language through single or multi-dimensional matrices of numbers could be a distinct source of the difference that machines bring to the reading and writing task. We believe we can learn from machines here.

<sup>8</sup> <https://talktotransformer.com/> (accessed: November 16, 2021).

## 2.4. GitHub

GitHub is a collaborative cloud code repository that allows our team to work together on the project and share with others. Using git revision control system along with GitHub provides granular access to every revision made to the project and makes it easy for several developers across wide geographies to contribute to the project.

We also use GitHub as a platform for scholarly communications, leveraging the built in static site generator included with github. Building on another project from Iliff's Experimental Humanities Lab, we have used our template for digital projects on GitHub to scaffold the repository for this project.<sup>9</sup> GitHub as a scholarly communication framework allows us to share our research and invite collaboration through the readability of our python code base itself, through careful commenting of the code, and through short pieces of web writing to reflect on the project and process.

## 3. Data Collection

As with all machine learning, data is critical. Remember, our task at hand is to train a model to generate commentary on a given passage of bible. Our dual hypothesis is that this model will be able to 1) generate useful text that will fit the accepted commentary tradition and 2) inject novelty into our understanding of bible. In a sense, this is a competing optimization task. We want our model to sound like bible commentary and at the same time to introduce new perspectives or approaches to these bible corpora.

To accomplish this competing optimization task of similarity and difference, we are collecting two related but different data sets to use in training our model. Again, thanks to the advances in generic pre-trained language models by labs at Google and OpenAI, we do not have to start from scratch with our model. Instead, we begin with a very capable generic language model in gpt-2. This generic language model can produce bible commentary without any additional training, yet two types of training will help make our model better.

### 3.1. General Knowledge about the Discourse

First, we are collecting as broad a set of general knowledge about bible and the discourse surrounding it as we can find. We can use sources such as wikipedia entries

---

<sup>9</sup> See <https://github.com/iliff/digital-dissertation> (accessed: November 16, 2021).

related to bible, social media posts discussing bible, popular and academic publications reflecting on bible, and historical commentaries on bible even back to antiquity. Ideally, this broad general knowledge data set would include perspectives from several different regions, cultures, people groups, and traditions. At this stage in the project, we have not had the resources to gain access to many of these generic data sources, but it is a critical part of our project roadmap.

### 3.2. Specific Knowledge to the Task of Bible Commentary

Most machine learning models perform best when tailored toward a specific task. Taking a bible passage as input and producing commentary on that passage from a broad background is a particular kind of text generation. So, in addition to tuning gpt-2 to the general discourse of reflections on bible, we are cultivating a data set of structured commentary on bible passages to help gpt-2 learn more about the specific task of producing bible commentary.

We have had several discussions as a team about what constitutes commentary on bible as well as where the boundaries are between bible and bible commentary.

To keep our early phase of this project very focused, we have chosen to work with the New Testament writing of Revelation and commentary on it from traditional Christian sources that are freely available online and limited to the English language. None of these initial narrow filters need remain for later stages of the project. We chose to focus on Revelation because it itself can be seen as a kind of commentary on much of the bible corpus and its language lends itself to the creative narratives sometimes generated by early versions of trained language models.

Our initial task specific data comes from the SWORD project of The Crosswire Bible Society.<sup>10</sup> Their list of English commentaries includes mostly public domain commentaries that are dated.<sup>11</sup> We have been granted access from the United Bible Societies to use their Translator's Handbooks, which provide highly specific commentary on bible passages related to translating these texts around the globe. We have not yet been able to process these handbooks to incorporate them into the model at this time.

We are distinctly aware of the limitations caused by our data collection decisions at this stage of the process. Finding openly available and machine readable corpora is always a challenge, particularly in a discourse that has been historically dominated by

---

<sup>10</sup> See <http://crosswire.org/sword/index.jsp> and <http://crosswire.org/> (accessed: November 16, 2021).

<sup>11</sup> See <http://crosswire.org/sword/modules/ModDisp.jsp?modType=Commentaries> (accessed: November 16, 2021).

institutional structures. We are open to suggestions for data sources we might utilize, particularly sources that we can easily convert to machine readable text.

## 4. Data Processing

### 4.1. Loading Texts

As mentioned in our data collection reflections, we have initially limited ourselves to task specific knowledge from the SWORD project. Part of the reason for this is that SWORD and Crosswire have made it easy to access their texts in machine readable fashion through a customizable command-line interface called diatheke and texts encoded in OSIS XML, one of the most common XML standards for texts related to bible.<sup>12</sup>

Arguably the largest digital collection of bible corpora in the world is the Digital Bible Library (DBL). Interestingly, the DBL has not opted to use OSIS as their XML standard. DBL uses an XML standard called USX, which is based on the older USFM schema.<sup>13</sup> We have not been able to get licensing from DBL at this time to use any of the corpora it contains beyond what is already available in public domain. We are hoping to get access to more DBL resources at some point.

The standard XML format of the SWORD commentaries and the customizability of the command line interface allowed us to write a parser to read any SWORD commentary and split the text into 3 columns: bible citation, text of citation, text of commentary. See our `diatheke_parser.py` file in the repository for this project to see the code for this parser.<sup>14</sup> Here is a sample:

```
text_list = [x for x in text_list if x not in ('', None)]
citation_text = []
for index in range(0, len(text_list)):
    if index % 2 == 0:
        try:
            citation_text.append([text_list[index].strip(), text_list[index +
1].replace('\n', '\t')])
        except IndexError:
            pass
for line in citation_text:
```

<sup>12</sup> See <https://wiki.crosswire.org/Frontends:Diatheke> and [https://en.wikipedia.org/wiki/Open\\_Scripture\\_Information\\_Standard](https://en.wikipedia.org/wiki/Open_Scripture_Information_Standard) (accessed November 16, 2021).

<sup>13</sup> For USX, see <https://ubsicap.github.io/usx/> (accessed: November 16, 2021).

<sup>14</sup> [https://github.com/iliff/theologiesofthedigital/blob/master/diatheke\\_parser.py](https://github.com/iliff/theologiesofthedigital/blob/master/diatheke_parser.py) (accessed November 16, 2021).

```
try:
    text = SCRIPTURE_DICT[line[0]]
except IndexError:
    print('line 61')
    print(line)
    pass
citation_scripture_comments.append([line[0], text, line[1]])
citation_text = citation_scripture_comments
```

## 4.2. Tokenizing

Computers work better with numbers than letters or words, so the next step of our data processing is to tokenize and encode all of the text we will pass to our model. Tokenizing simply means splitting a text up into smaller parts (i.e. tokens). Tokens can be letters, words, parts of words, punctuation, etc., depending on your task and the tokenizer you use. With some slight modifications, we use the GPT<sub>2</sub>Tokenizer to split up the verses and the commentary in our data into smaller bits that correspond to the 50,000 word vocabulary of gpt-2. Each item in this vocabulary has a corresponding integer value, so when we tokenize our text we also encode as integers.

Before we tokenize and encode, there is some cleaning up to do, such as removing unwanted characters, removing duplicate comments, and trimming out anything else in the data that will just add unwanted noise. This cleaning step may seem mundane, but it is often one of the places in a project where certain assumptions about the dataset are enacted. So, we need to be transparent about the task involved in the data preparation and always question them as we learn more about how the model uses the data. At present, here are the things we do to clean our dataset before passing it to the model:

```
print('cleaning df ...')
df = self._clean_df(df)
if max_df_len:
    df = df.sample(frac=1.).iloc[:max_df_len]
    df = self._add_sequences_to_df(df)
    df = self._add_sequence_lengths_to_df(df)
    df = df.sort_values(by=['total_token_length'], ascending=True)
return df
```

## 4.3. Preparing Data

Once our texts are tokenized and encoded, we construct a dataframe, which looks much like a spreadsheet. Our dataframe has one column with the encoded version of a verse from Revelation and one column with the corresponding encoded version of the commentary on that verse. We have as many rows in the dataframe as we have



samples of verses and commentary from the commentaries we pulled from SWORD using the diatheke interface. We could add many more rows to this dataframe as we increase the commentary data sources we can incorporate into the model.

#### 4.4. Model Training

We are taking a supervised approach to training this bible commentator model. This means that we begin with the gpt-2 pre-trained language model, which is a neural network with several layers depending on which size of the model we use.

OpenAI has released their X-Large gpt-2 model, which has 1.5 billion parameters and 48 layers. Jay Alammam's piece, "The Illustrated GPT-2," provides some nice visualizations of the different sizes of the gpt-2 model.<sup>15</sup> Since the X-Large has not been available, and due to the demand on computing resources and the design of our model, which trains 2 gpt-2 models in parallel, we have focused on the large and medium gpt-2 models, which have 32 and 24 layers respectively.

With our supervised approach to training, we construct a dataset that has two inputs ( $X_{\text{verse}}$ ,  $X_{\text{commentary}}$ ) and a known output ( $y$ ). Because we are using gpt-2 to generate text, one word at a time, from a seed prompt, here is what these inputs and outputs look like:

```
X_scripture = an integer encoded sequence of the entire verse from Revelation
X_commentary = an integer encoded sequence of the commentary at each stage (this
               will grow each pass)
y = the next word (encoded sequence) of the commentary
```

Every pass through gpt-2, our model uses these  $X$  inputs to predict the next most likely word in the commentary and appends this word to the  $X_{\text{commentary}}$  input for the next training pass. Since this is supervised learning, the word that the model predicts ( $y^{\wedge}$ ) is compared against what we know to be the next word ( $y$ ) in the commentary we are using as our training dataset and a loss is calculated based on the difference between  $y$  (actual next word) and  $y^{\wedge}$  (predicted next word). Using this loss calculation, the model goes backward through its layers and adjusts the weights of each connection in the network before it runs through the next pass. To make our training a bit more

<sup>15</sup> See <http://jalammam.github.io/illustrated-gpt2/> (accessed: November 16, 2021).

reasonable to manage, we actually only adjust the weights of the edges of the network after approximately 16 commentary samples have been processed.<sup>16</sup>

We have defined our commentary length output to be 151 words, so we ask the model to make this many predictions for each verse input.

## 5. Adding Knowledge to our Training

To provide more targeted and nuanced training for our model, we are experimenting with a more complicated training process that passes an additional input to the model indicating a best fit generic background knowledge sequence for the commentary we are training on. So, in addition to  $X_{\text{verse}}$  and  $X_{\text{commentary}}$ , we will pass the model  $X_{\text{tfidf}}$ , an encoded sequence representing a passage from a more generic corpus related to Revelation. The reason we call this input  $X_{\text{tfidf}}$  is that we use a similarity based linear model to find relevant knowledge passages related to a given commentary sequence and then we use term-frequency-inverse-document-frequency (tf-idf) to pick the most informative passage from the list. For more details on this additional training aspect, see the `CPULinear` class in our `generator.py` file.

We are hoping this additional knowledge input will provide better intuition for the model to generate text that fits the commentary discourse without over-fitting to that discourse. This is one strategy we are employing to push toward our dual optimization task of fitting the discourse and introducing novelty.

## 6. Initial Results

We will add more to our discussion about our training process for the model and the model design, but for now, we want to share some initial results produced by the model so far.

### 6.1. General Tuning of the Language Model

After a few rounds of training (more specifics to come on exact durations and conditions of these training runs) on our bible commentary corpus to focus the general knowledge of our language model on our desired discourse, we can see the language

---

<sup>16</sup> For an excellent vizualization of the intuition around this backward process of a neural network, where weights are recalculated, see 3Blue1Brown's excellent video: What is backpropagation really doing? | Deep learning, chapter (<https://www.youtube.com/watch?v=Il3gGewQ5U> [accessed November 16, 2021]).

model tuning toward the biblical discourse. To remind us, when we give gpt-2 a prompt, it takes the prompt and then uses its language model to choose the most probable next word that would appear and writes it. Then, it takes the prompt plus the first word it chose and does the same process again to choose the next word, and so on, until we tell it to stop.

If we pass the phrase “Artificial Intelligence revealeth” to gpt-2 without any additional general knowledge training, here are the kinds of comments gpt-2 returns (each bullet is a separate attempt at a response):

**artificial intelligence revealeth: (generic gpt-2)**

- its hidden potential – can change everything [9 December.] ‘Robofication: the second age of technology’ – how advances technology as soon [...] Posted
- what humanity is truly and as much potentials. – Aeswarya Mhamisundre Rao I have learned today what people cannot possibly
- a huge amount that most men still consider secret: it knows who my partners sleep with because I’ll read “The Artless Sin by John Fulk
- us that no more be in our way before, it could do that with ease in every region;
- To our new technology: be ye patient until
- by the year 1040. On an ordinary afternoon they get only 7 pdays - 10p nights (for every 7+2x7 days the
- “who do-they know”- and not why.” The new approach to research that AI represents may create “big implications from technology design (technology safety
- not my opinion to know the opinion  
2 John 5th & 6c The kingdom must always rest as a safe deposit; but there was one good
- the source of its madness! An unknown faction which has seized control over their galaxy now requires no loyalty nor restraint before deploying them on all available fields as
- (in 2 seconds) 1 3 50 1 The most incredible revelation a noncomputer system may give in one minute 7 16 944 This article in a
- : to detect through surveillance intelligence : technology for a machine designed after a very closely aligned brain mechanism. “I didntl say, only an advanced

- ; thereis none, except such intellect and moral faculties created since all man know thereof to which knowledge cannot comove; it also teachst that mankind
- . From his earliest form of knowledge the Creator sent them with his plan - that they know so clearly why everything He works (everything). Even this,
- the God within? Are the people we need at prayer with god a danger, do I do harm through following blindly after someone who calls him “Jesus
- you how not, an argument for creating “the man of clay which ye will make your prophet”: as the word means man- of metal who by

There is a wide range of concepts and vocabulary covered here, with some attempts being more coherent than others. Interestingly, we see some biblical language appear in some of the comments, even without any tuning toward the discourse. This could be a result of the KJV style “revealeth” we used in the prompt or it could indicate that talk about artificial intelligence sometimes overlaps with talk about religion and bible and related things.

Providing the same prompt to our trained gpt-2 model, here are the kinds of comments we get:

**artificial intelligence revealeth: (trained gpt-2)**

- the life: Who have shown his true integrity?
- no truth with my nation a fief;
- nothing else than Jesus.
- these truth things as all wisdom came upon all things which is done for such
- our works upon one good God Jesus for the grace made known which thou went down,
- great many by our good alightsomenes the faith?
- great men the chief pleasures among Christ.
- every scripture.
- to ourselves that our things is profitable to each creature for to teach in our teachers for we received. we.
- him no harm: but we can save as ye save the weak sheep which perish against such,

It is clear that our trained language model is sounding more like biblical discourse, perhaps even a little too much. As we stated in the problem definition, we have a competing optimization task of building a model that can write in the style of bible commentary, but also introduce novelty. We are clearly not there yet. One of our strategies for better tuning this general knowledge of the language model will be to incorporate less explicit and more contemporary reflections on bible. We plan to incorporate some Wikipedia data as well as some broader reflections on bible and Revelation in particular from current scholars such as Timothy Beal. Additionally, many of the comments on bible passages actually restate the passage itself, so we will be working to balance the weight of the biblical texts on the learning of the model.

## 6.2. Specific Task of Commentary

At this early stage of testing and training, we clearly have many adjustments to make, but we do have some examples of the trained models attempts to comment on a biblical passage. Let's consider a few examples.

**Passage Prompt:** And when he had opened the fourth seal, I heard the voice of the fourth beast say, Come and see.

**Model Commentary:** He said to them. The fourth of his beasts came forth (Mr 7 and his two other two angels) and his angels, and their dragon (Hebrais and Hippoleon. And as it is in iTi 4, 1) were made to stand before them as it is proper for the third beast in the world

What I find interesting about this commentary is that the model is trying to cite other passages of bible with the parenthetical phrases. Perhaps “Mr 7” is an unknown lost text from antiquity? We have seen in other contexts that the way gpt-2 learned how to handle numbers is not always coherent. Yet, “iT i 4, 1” looks a lot like a reference to 1 Timothy 4:1, which does refer to future times. Is the machine suggesting an intertextuality that we may not have imagined before?

When I passed this exact prompt to the standard gpt-2 model using Talk to Transformer, each time the model tried to keep writing bible. Here is an example output from the untrained model:

**Prompt:** And when he had opened the fourth seal, I heard the voice of the fourth beast say, Come and see.

**Continuation**

When I saw the fourth seal, I knew – the seal of the living God, who was, and is not, and has not been, and is to come, the Most High, who is seated on His throne, and coming with judgment to judge the quick and the dead.

I saw the dead, great and small, stand before God.

“And the dragon was in the midst of them, and he had a great millstone round his neck, that he might crush those who were born of women and who are born of iniquity: and they were given up to destruction; and death and hell were given up to her.”

—Revelation 21:11-16

The standard gpt-2 model is designed to continue the prompt with the next most probable word. One of the design items we have added to our model is to append an “end of line” character after the prompt to indicate to the model that continuation is not desired, but rather reflection on the prompt. This raises two pertinent questions:

1. If we trained toward continuation, could our model write bible passages that were comparable to what we would find in those corpora accepted as bible?
2. Are there ways in which bible commentary or interpretation are better conceptualized as continuation of rather than reflection on?

One last example of our early model’s performance:

**Passage Prompt:** I am Alpha and Omega, the beginning and the ending, saith the Lord, which is, and which was, and which is to come, the Almighty.

**Model Commentary:** So often a Christian. Who were chosen as of the gospel in Jesus before Jesus when this was the most Dangerous – whose kingdom came to him according to his preselection.

Perhaps “so often” is a summary of “is, was, is to come”? But, much of this commentary is coherent with chosenness and preselection and gospel and kingdom and Jesus. What might the “Dangerous” signal in this passage? Here I think we have a seed of what we are hoping from our experiments in reading with machines. There is much familiar in this machinic reading of Revelation 1. Is this a reading that brings together the bookend “Alpha and Omega” passages of Rev. 1 and Rev. 22? Does it offer us anything new to reflect on?

We will continue to run experiments in training and designing this model to achieve our competing optimization task of building a bible commentator that fits the discourse but introduces novelty. As we continue this project, we will continue to explore what our machinic partner might teach us about reading bible in a digital age.