Generative algorithms to improve mental health issue detection

Jimmy Lam[1] , **Willem-Paul Brinkman**[2] , **Merijn Bruijnes**[3]
TU Delft

Abstract

Schema therapy is a physiological treatment technique for mental health issues. Based on the thoughts and behaviour, patients are classified to a schema mode which represents their current state of mind. Automatically classifying these thoughts and behaviours could improve detection of potential mental health issues as well as provide better and faster recovery. This research attempts to effectively generate schema-based stories that would be used to train machine learning models such as Support Vector Machines and Recurrent Neural Networks to classify stories from patients about their daily experiences. Experimental evaluation using the OpenAI GPT-2 model shows that it is possible to generate correct and coherent stories with a minimum of 58.7% correctly classified samples even with sub-optimal data. Using conditional prefixed queries, the OpenAI GPT-2 model can generate stories that resemble the given data but with little to no similarity in terms of BLEU scores.

1 Introduction

1.1 Motivation

Mental health issues are prevalent on all sides of human society. It is estimated that at least 10 percent of the world's population is affected by mental, neurological or substance abuse disorders [7]. One way of detecting mental health issues is through schema therapy which is a technique used for recurring long-standing problems [23]. Patients are assessed based on the way they act and interact with others as well as how they feel and think about each other. Their behavior is then classified using distinct schemas representing the emotional state they are in. Early and correct determination of the patient's ailments can lead to better treatments and faster recoveries [9]. One way of digitally diagnosing patients was introduced by Allaart where he let people converse with a chat bot about their thoughts and feelings. The chat bot would ask specific questions to the patients about their stories and ask them to rate their emotional state based on a questionnaire. One of the areas where the author proposed future research was to use the chat bot to try to predict the schema which would belong to the patient. He proposed that with more sentences and phrases, the model could predict schemas more accurately [1]. To gather all these stories takes a lot of resources and the information collected is regarded as sensitive information. In an effort to gather more stories while at the same time protecting the privacy of patients a study was done to use generative algorithms to create them.

1.2 Research questions

How well can a generative algorithm (e.g. RNN based encoder-decoder network) write stories that fit specific schemas?

- Is there already research done about algorithms that generate stories and which techniques are the most effective ones?
- What does an implementation of an algorithm look like that can generate stories according to a given schema?
- How similar are the generated stories compared to the data set and what techniques should be used for comparison?

1.3 Approach

The main research question was split into three sub-questions each going into more context in order to answer the main question. The first sub-question addresses the problem of generating text that might already have been done by others. A literacy study was conducted to find the most common generative algorithm techniques for text generation and how they are used. Comparisons amongst different techniques give insight into which technique has the highest probability of success when answering the main question. It also gives insight into the data, data-formatting and data-processing needed for the implementation of such an algorithm.

The second sub-question revolves around the creation of the generative algorithm using the best technique found in the first sub-question. This part of the paper delves into the specifics of the algorithm and the technologies used during the execution. The aim was to demonstrate a reproducible algorithm that uses open-source resources. The algorithm has to be able to generate similar stories by either a given schema or given a data set of stories where all the stories have the same schema.

After generating the stories it is important to evaluate the results and compare them to the real stories. This comparison is important as stories generated should have as little correlation as possible with the original data set. The reason for this is that the privacy rights of the patients will be compromised if the generated stories are too similar to the real stories. The accuracy of the stories plays a pivotal role in the determination if generated stories are a viable option to train machine learning models. Using manual evaluation the stories were assessed if stories had a consistent theme/ topic and if they corresponded to the correct schema.

This research is a continuation of the research done by Burger and Allaart [6] [1]. Burger's aim was to test whether it was possible to assess stories using machine learning techniques. Burger took three types of learning algorithm types (KNN, SVM and RNN) and showed, with no emphasis on performance, that it was feasible to classify stories using schemas. Based on this research it can therefore be concluded that the accuracy of generated stories can be compared to stories that have already been assessed. Allaart showed in his research that determining a patient's schema mode is possible using a conversational agent in the form of a chat bot. During his research, Allaart collected more than 1.100 messages using the Amazon MTurk questionnaire platform. Allaart asked the participants to tell the chat bot a story about something that happened to them recently and asked them to rate the following schemas *[Happy, Sad, Angry, Detached, Impulsive, Vulnerable, Punishing, Healthy]*, on a scale from 1 to 6, about their current state of mind. This data was used for the generation of the stories.

2 Overview of generative algorithms

To broaden the horizons and introduce more context about the topic this chapter of the paper revolves around the different types of generative algorithms. A variety of internet resources such as research papers, programming articles and e-books were consulted as part of the literacy study. Furthermore, this chapter delves into the workings of each algorithm, what their characteristics are, differences and similarities amongst them and what the advantages and/or disadvantages are.

2.1 OpenAI GPT

GPT-1

OpenAI is an artificial intelligence research company that specializes in developing and researching artificial general intelligence for the benefit of all of humanity. In 2018 researchers at OpenAI noticed that while unlabeled text can be abundantly found on the internet, it is not the case with labeled data. They argued that it is challenging for discriminatory trained models to perform well when provided with scarce labeled data. This led to the development of the GPT model which aimed to improve the performance of language models. GPT uses generative pre-training followed by discriminative fine-tuning before it is actually used for NLP tasks. Using semi-supervised learning, the researchers pre-trained the model with a data set of 7.000 unique unpublished books from a variety of genres. This was followed by supervised fine-tuning where the model was tested on natural language inference, question answering and commonsense reasoning, semantic similarity and classification. GPT was able to perform nine out of twelve tasks better than similar machine learning models with absolute improvements of 1.5% on textual entailment, 5.7% in question answering and 8.9% in common sense reasoning [18].

GPT-2

In February 2019 OpenAI announced that a successor of the GPT was developed named GPT-2. The main difference between the original model and its successor is that GPT-2 is a direct scale-up of GPT and therefore had four different versions. Each version varied the amount of parameters they had and all were trained on 8 million web pages. One of the new additions to GPT-2 was that the model performed zero-shot NLP tasks. Zero-shot NLP tasks consists of reading comprehensions, translation, summarizing and question answering on data that the model has not seen before. This made it possible for GPT-2 to do NLP tasks on domains it was not trained for. OpenAI was able to show that GPT-2 has state-of-the-art performance on seven out of eight zero-shot NLP tests. [19]

2.2 Generative adversarial network (GAN)

A generative adversarial network is a machine learning technique where two neural networks are pitted against each other in a zero-sum contest to optimize the output data. First introduced by Ian Goodfellow in 2014 as a means to generate arbitrary images without the help of humans. GANs are made up of a discriminator network and a generator network. The discriminator network is in charge of distinguishing generated samples from the real samples and the generator network is in charge of generating realistic samples that could fool the discriminator into being real samples. Each network will then be equipped with a value function that gives a numeric representation of the model's performance. The contest then becomes a minimax game where the goal is to maximize the value function of the discriminator and minimize the value function of the generator. This leads to a situation where the real distribution of data is replaced by much of the generated distribution [10].

GANs have shown to be reliable machine learning technique used for various tasks in different fields of computer science. Although used most often in image generation, GANs has also shown that it can be applied for upscaling low-resolution textures in video games and improving astronomical imagery by simulating gravitational lensing for dark matter research [14] [22].

2.3 Recurrent neural networks (RNN)

In 1982 John Hopfield discovered an associative neural network which he called Hopfield networks. The Hopfield network was the first neural network to include recursive characteristics, combined with storage and binary systems. This model would later evolve into the recurrent neural network that is known today. RNN's is a neural network class where connections between nodes make up a directed graph that shows similar behavior to human brains. The nodes in the model form an internal memory which is used to predict the output based on the data the model was given. At the end

of each prediction round, a loss function is kept that will be propagated back through the model and adjust the weights of each node. By reducing the loss function on a certain criterion the model is able to maximize its predictions [4].

2.4 Model use cases

Researchers at the University of Toronto have shown in a 2019 paper that it is possible to use GANs for text generation. By transforming text samples into numerical vectors the researchers showed that the models were able to generate coherent and diverse sentences based on a variety of data sets. Using the standardized BLEU score test the researchers have shown that GANs, on large-object based data sets, are able to produce sub-human results [5]. The researchers used the COCO data set during training and testing of the model [8] which contains 1.5 million objects and 330.000 images of which 200.000 are labeled. This leads to one of the drawbacks of using GANs for text generation as the model requires a lot of different types of data which have to be verified frequently during training of the model for accurate predictions.

In a 2020 paper from the German Aerospace Center, Sivasurya Santhanam showed using the *the Lord of the Rings* data set he was able to generate context-based stories using a RNN. The network was fed keywords and generated sentences in the same context. The objective of the research was to measure the similarity of the generated text in relation to the context of the keywords. The author used the nouns of each generated sentence to define the context of the sentence. Using the cosine similarity the author measured if the generated sentences shared the same context as the given keywords. The author concluded the research with a cosine similarity of 67 percent to 85 percent for generated sentences in relation to the context given. In the discussion, the author mentioned that the paper did not include sentence coherence in the research as it was out of the scope of the paper. He observed that the model was overfitting on the data and could therefore not generate grammatically correct sentences. Overfitting happens when RNNs get trained too well on a particular data set and can not generalize properly on new data. The model memorizes noise and randomness as opposed to the underlying patterns the model is supposed to learn. This causes the model to make predictions based on the randomness [20].

In 2020 OpenAI GPT-2 was used for text generation on a Chinese dataset named BaiduBaike and LLKT. The researchers mentioned the diversity of sentences made it impossible for automatic evaluation metrics. They opted for manual evaluation of the generated data and found that OpenAI GPT-2 could generate text very similar to human-produced texts. Furthermore, they concluded that the model showed signs of repetition in the generated sentences due to shortcomings in the data provided for fine-tuning. Possibly due to inconsistencies in the writing style and length of training sentences and no standardization of the language rules in the data set [17].

2.5 Comparison of the generative algorithms

While all model use cases show promising results when used for text generation, one common problem is prevalent across all of them. The performance of the model is very dependant on the quality of the data set. Goodfellow found in his paper that GANs get BLEU scores close to 1, which means that the text is very similar if not identical to the original input. The minmax policy of the model causes sentences identical to the input but with one or two differences to be accepted [10]. This makes GANs not suitable for this research as the generated stories should have the least amount of similarity with the input as the privacy of the authors of the original stories would be at risk. RNNs tend to generate incoherent sentences, which would result in algorithms that learn to classify stories based on incoherent sentences. As RNNs have to learn grammatical structures and rules from scratch, it is common for these models to require large data sets. For this research the OpenAI GPT-2 model was chosen as OpenAI suggested in their *Language Models are Unsupervised Multitask Learners* paper that fewer data can also lead to promising results [19].

3 Implementation

Woolf made in his 2019 article a Google Colaboratory file that allows users to fine-tune OpenAI GPT-2 on any data for text generation. Originally the code was used to imitate certain users on Twitter by generating tweets based on their user history [21]. This research used Woolf's Google Colaboratory file in conjunction with the official OpenAI GPT-2 source files as the file only supported the unconditional operating state. OpenAI GPT-2 has an additional conditional operating state which could be found in the official OpenAI GPT-2 source files. To have a more thorough analysis on OpenAI GPT-2's text generation both operating states needed to be tested. Due to a software bug in the code from the official OpenAI GPT-2 source files, it was not possible to encode the fine-tuning data to the correct format for the model. However, this software bug was not prevalent in the Google Colaboratory file and therefore the encoding was done with Woolf's code while the generation was done using the official source files from the OpenAI GitHub [15].

3.1 Conditional and unconditional algorithms

The conditional operating state allows the model to take in the additional text before generation. The model forms a sentence starting with the prompt that was given,

for example:

prompt = "I had a wonderful day today because"

result = "I went to the zoo with my friends."

The unconditional operating state does the opposite and generates text based on the data it was fine-tuned on.

Algorithm 1 Unconditional OpenAI GPT-2 text generation

1: **procedure** GENERATE
2: gpt2 $\leftarrow download(OpenAIGPT-2)$
3: fineTuneData $\leftarrow import(dataset)$
4: $gpt2.finetune(fineTuneData, steps)$
5: $gpt2.generate(samples, top_k, temperature)$

Algorithm 1 represents the unconditional pseudocode used for this research. The algorithm first downloads the model

from the OpenAI server which is followed by an import of the data set that is used for fine-tuning. Then the model is instructed to fine-tune itself for any given amount of steps and to generate a given amount of samples using two parameters [15].

Algorithm 2 Conditional OpenAI GPT-2 text generation

1: **procedure** GENERATE
2: gpt2 $\leftarrow download(OpenAIGPT-2)$
3: fineTuneData $\leftarrow import(dataset)$
4: $gpt2.finetune(fineTuneData, steps)$
5: $input(prompt)$
6: $gpt2.generate(samples, top_k, temperature, prompt)$

Algorithm 2 represents the conditional pseudocode used for this research. This algorithm is similar to the unconditional algorithm but the algorithm will ask the user to input a prompt [15].

3.2 Parameters

OpenAI GPT-2 accepts two parameters during the generation of text namely temperature and top_k. Temperature relates to the amount of variability the model can have on the fine-tuning data and pre-trained data. High values cause the model to take more inspiration from the pre-trained data whilst low values cause the model to predict sentences more based on the fine-tuning data. Top_k relates to the number of guesses the model can make on its predictions. OpenAI GPT-2 produces at any given time a variety of results from which it picks the best one. When top_k is high the model limits the number of guesses it makes while low values allow it to do the opposite [15].

3.3 Data partitioning

Allaart's data set was a csv-file that contained all the stories and their respective schemas. Stories that were rated with a 3.5 or higher on a specific schema were assigned that schema [1]. This research will build upon this notion and partition all the stories according to their assigned schemas. Table 1 shows what the result is after partitioning and how the data sets relate to each other in terms of size with the *is_healthy* data set being the largest and the *is_impulsive* data set being the smallest.

Data sets	# tokens	# entries
is_vulnerable	122.930	350
is_healthy	377.901	1125
is_angry	150.393	436
is_impulsive	62.464	201
is_detached	131.370	376
is_punishing	82.822	238
is_happy	298.993	871

Table 1: Representation of the data partitioning of David Allaart's data set.

3.4 Pre- and Post-processing

The participants in Allaart's research were not limited in the things they could say to the chat bot. This resulted in text that does not suit the context of this research. Therefore the text was pre-processed using the following rules:

- Lower-casing of all text
- Replace misspellings, contractions and grammar mistakes
- Add missing phrases, end marks and comma spaces
- Delete certain response words, e.g. OK, Yes, No, Quit, Good Bye
- Enquiries on how to exit the chat bot conversation
- Comments regarding the functionality of the chat bot

In *A text generation and prediction system: Pre-training on new corpora using bert and gpt-2* the author mentioned that text generation using OpenAI GPT-2 leads to duplicates of the data it was supposed to learn from [17]. Assessment of the inherent qualities of duplicates does not bring meaningful results in the context of this research. Another notable observation from the generated samples were that the model tends to overestimate or underestimate its generation. Overestimation happens when the model stops mid-sentence as it reaches its required length and underestimation happens when the model pads its output with additional words or phrases. Likewise, unfinished sentences introduce phrases and sentences which do not add any additional context to the story. Both behaviors were analyzed further in chapter 5. To combat both of these behaviors, additional post-processing was applied which consists of the same actions taken in the pre-processing, in addition to two new rules:

- Entries which are identical to any entry in the fine-tuning data
- Preemptively removing unfinished sentences and/or phrases after the last full stop

4 Evaluation and Results

An integral part of this research was evaluating the results found during the experimentation phase. This chapter of the research delves into how the tuning of the parameters of the model led to different results. As well as how well schemas can be fitted to a generated story and how many similarity the generated stories have with the original stories.

4.1 Generated data set

According to Woolf's article, it was recommended that values for all the parameters should stay within a certain bound. These bound represent the trade-off the model makes in its predictions based on the fine-tuning data and its pre-training knowledge. If the parameters tell the model to correspond too much to the fine-tuning data it will just copy the entries of the data set. When the model is told to correspond more towards its pre-training knowledge the results vary drastically, this leads to results that do not represent the fine-tuning data. Figure 1 represents the bounds while figure 2 and 3 show the

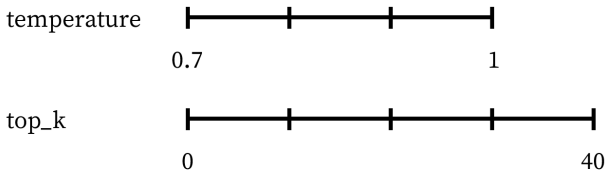

Figure 1: OpenAI GPT-2 parameter bounds

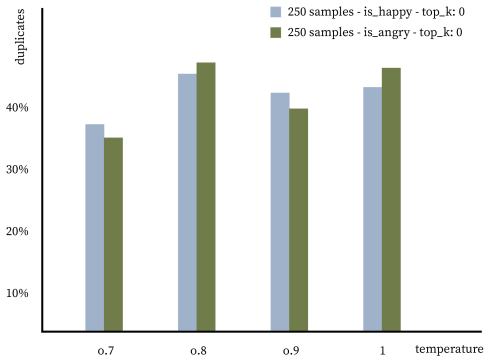

Figure 2: Number of duplicates using temperature

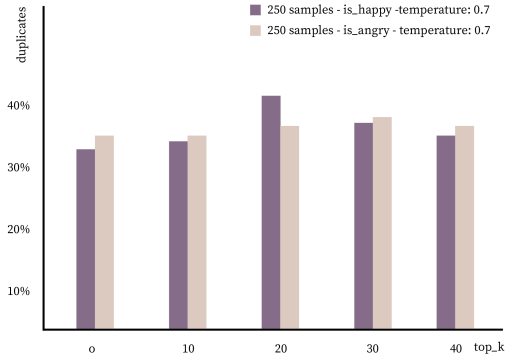

Figure 3: Number of duplicates using top_k

number of duplicates made by the model upon changing the values of the parameters.

The values 0.7 and 0 for temperature and top_k respectively produced the least amount of duplicates. Using these parameters a total of four data sets were generated, a conditional data set and an unconditional data set for both is_happy and the is_angry schemas respectively. To test whether stories generated using different prefixes would yield a difference in their characteristics, only one prefix was used for the is_happy stories while for the is_angry stories a total of four prefixes were used. For each is_happy sample the prefix: "I had a wonderful day because" was used and for each is_angry sample an arbitrary prefix from the following list was chosen: ["I have to fight because", "i am furious at someone", "i want to punish people for", "i feel treated unfairly because"]. All prefixes were based on the intent list Allaart used for schema classification in his research, see Appendix A. These data sets are the same size as the is_happy and is_angry partitions from Allaart's data set and were used for the schema, coherence and independence assessment [1].

4.2 Schema and coherence assessment

Using Allaart's intents list it was possible to analyze when stories belong to a certain schema, see Appendix A. For example, when a story contains elements such as spontaneously, calmness or a sense of connection with others the story shares many intents with the happy child schema. Allaart's assessment of stories was done in the same manner where he submitted a small subset of the collected stories to an expert panel. The experts were asked to rate the submitted stories on which schema they thought the stories belonged [1]. To measure inter-rater reliability Allaart used Cohen's Kappa where the ground truth was the score of experts. The first author also assessed the stories submitted to the expert panel which resulted in the Cohen Kappa matrix found in table 2.

	Ground truth	Expert 2	Lam	Expert 1
Ground truth				
Lam		0.63		
Expert 1	0.75		0.7	
Expert 2	0.58			0.65
Average	0.66	0.63	0.7	0.65

Table 2: Cohen's Kappa matrix

Research by Han, Zhang and Park concluded that using machine learning techniques to classify Allaart's data set gives accuracy results below chance [12] [24] [16]. A manual approach was therefore chosen to assess the performance of the model in terms of its correctness and coherence. The reliability of manual labeling was measured between the experts and the first author who did the same manual assessment as the experts [1]. Per schema, a statistical sample with a confidence level of 90% with an error margin of 10% was taken from the conditional and unconditional generated data sets.

The work-flow for rating generated stories was done according to the following framework:

Step 1: Correct any grammatical errors.

Step 2: Rate the coherence of the story using a score from 1 to 6. Where the score of 1 means that the story starts and finishes with a completely different schema intents and the overall topic changes. A score of 6 implies the opposite as the story belongs in its entirety to one intent and the topic remains the same throughout the story.

Step 3: Classify the story using a score from 1 to 6 on how likely it belongs to one of the possible schemas using the intent list provided in Appendix A. A score of 1 means that the story shared no intents with the schema from which its generation is based on while a score of 6 means that it shares more than 6 intents.

Each subsequent score equates to an intent shared with the schema the story is based on.

Generated happy story 1

"i had a wonderful day today because my dads health was good, it lifted my spirits and i felt calm after a few days. i would say i was the happiest person i have ever been in a few days because of all the support i had received and i feel grateful to him for"

Example:

Step 1: Story did not contain any grammatical errors, so proceed to step 2.

Step 2: Story starts and ends with the same intents and does not diverge halfway through the story and therefore gets a 6/6 score for coherence.

Step 3: Story talks about a person that is calm, happy, supported and grateful for the presence of others around him or her. This means the story shares four intents with the is_happy intents list and therefore got a 4/6 score for correctness.

	Conditional		Unconditional	
	is_happy	is_angry	is_happy	is_angry
Samples	63	59	63	59
C1 + C2	37	40	4	6
I1 + C2	9	7	3	4
C1 + I2	2	3	45	42
I1 + I2	24	9	10	7

C1 = coherence, C2 = correctness, I1 = incoherence, I2 = incorrectness

Table 3: Generated samples manually scored

By imitating Allaart it can be said that if a story scores a 3.5 or higher for coherence or correctness then it will be classified as such [1]. Generally speaking, the algorithm was able to produce better results using a conditional prefix, see table 3. Furthermore, the model reacted positively to multiple prefixes as opposed to just one. The amount of incoherent and incorrect stories see a 14.1% relative decrease while the coherent and correct stories see a relative increase of 2.8%. Due to inconsistencies in the labels of Allaart's data for example, when participants rated their emotional state differently from the stories they told see figure 4, the unconditional version generated more incorrect stories. An example of this behaviour can be seen in figure 5 where the model was supposed to create an is_happy story but ultimately does not resemble any of the intents associated with the is_happy schema. Using a conditional prefix the model got an overall better coherence and better correctness presumably because the model was able to learn the writing style from the fine-tuning data while using its own pre-trained grammatical knowledge to finish the story.

4.3 Independence assessment

According to Celikyilmaz, Clark and Gao, one of the most common ways of evaluating similarities within generated and non-generated text are untrained automated metrics.

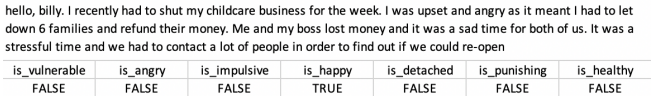

Figure 4: Example of incorrect classification

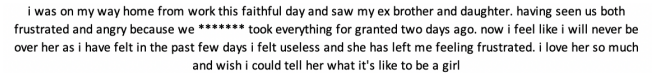

Figure 5: Example of incoherent generated story

Amongst the untrained automated metrics, the BLEU test is one of the most common and was used to test how the independence of the generated stories [2].

BLEU score

BLEU is an algorithm that was initially purposed for machine translation but has shown to be useful in comparing two texts. It takes a candidate text and a reference text and returns a value between 0 and 1, 0 meaning no match at all between the text, and 1 meaning a perfect match. BLUE uses a geometric mean of n-gram precision scores where n ranges from 1 to 4 which tells the algorithm on how many words it should calculate the score.

	Conditional		Unconditional	
BLEU	is_happy	is_angry	is_happy	is_angry
1-gram	0.16	0.09	0.2	0.12
2-gram	5.90e-155	4.06e-155	6.84e-20	1.24e-43
3-gram	4.80e-204	3.58e-204	2.22e-102	5.98e-93
4-gram	1.13e-231	8.84e-232	3.12e-112	9.27e-100

Table 4: BLEU scores for all four generated data sets

Table 2 represents the BLEU score of both conditional and unconditional generation for both schemas. The results were conceived by iteratively traversing each generated data set and calculating the BLEU score. Once calculated the results were added together and divided by the size of the data set to get the average.

Generally speaking, all BLEU scores were very close to 0, this means that almost all generated sentences show close to no resemblance to the fine-tuning data. Another observation made was that conditional prefixed stories show fewer similarities compared to unconditional stories. As the model was more restricted due to the prefix it was able to create more unique answers.

4.4 Notable observations

In the generated stories it was observed that OpenAI GPT-2 has a tendency to generate stories that are not fully finished, see Appendix B. The model was given a parameter that states how long the generated text should be. The model then either overestimating or underestimating its predictions for the next words. As the model tried to fulfill its criteria the generated samples were either padded by additional words or the model cut off its generation when the length of the story exceeded the criteria. One solution suggested by Grinberg was

by parsing higher values for the text length [11]. By tracing back the last full stop in every generated story and discarding the phrases and sentences that follow, allows stories to have a proper finish.

5 Responsible Research

For the research to be trustworthy it needs to be done ethically responsibly and easily reproducible. An effort was made to conduct this research in such a manner by presenting the results in an honest and straightforward way.

5.1 Scientific integrity

Many aspects of this research take inspiration from other research done by various scholars. References were at all times credited to the original authors as well as time-stamps on when the resources were used. Therefore, it is possible to trace back the sources whenever changes occur to the original reference or the validity needs to be tested. This research was part of a series of other research into similar topics. As such collaboration amongst the peer researchers was established to share results, data and techniques. The pre-processing of Allaart's data was done in tandem with the peer researchers as that was also part of their research [12] [16] [24] [3].

5.2 Reproducible

To ensure reproducibility, all the source code used was uploaded to the GitLab repository which was provided by the course CSE3000. During the research, Allaart's data was used which had been modified to suit the context of this research. The modified data was uploaded to the GitLab just like the source code that was used. As part of this research, a lot of data was generated which has an element of randomness to it since OpenAI GPT-2 might generate different stories every time it's tasked to. However, all of the data used and generated in this research has also been uploaded to the GitLab.

Link to the code: https://gitlab.ewi.tudelft.nl/cse3000/2020-2021/rp-group-43/rp-group-43-wlam

6 Limitations

OpenAI GPT-2 model comes in 124M, 345M, 762M and 1542M versions which represent the number of weights the model takes into consideration whenever making its predictions. The difference can be quite big as performance improvements in NLP-tasks can go up by as much as double the performance [19]. The purpose of this research was to show the possibility and performance of using a generative algorithm for the benefit of schema therapy. Therefore, only the 124M version of OpenAI GPT-2 was used for this research as it gave a sense of the performance possible using a state-of-the-art model. The results of this research might not be indicative of the OpenAI GPT-2 model as a whole as the amount of correct and coherent stories might be higher or lower depending on the version of the OpenAI GPT-2 model used.

As this research was a continuation on Allaart's research an emphasis was put on the data he collected. As shown in chapter 4.2 the model is susceptible to data that do did share the same sentiment. When a fine-tuning data set with incorrectly classified stories was given to OpenAI GPT-2 it learned to generate stories with different sentiments. The usage of other data sets might yield different results and conclusions.

The manual labeling of the generated story was done with a confidence interval of 90% and 10% error margin while the most common metric for analytical research is a confidence interval of 95% and 5% error margin [13]. Using the most common metric would result in 205 is_angry samples and 267 is_happy samples. A consensus was made for this research which results in a lower probability of the sample mean to be included in the assessed samples. In the context of this research, a lower confidence interval and higher error margin could lead to the assessed stories not containing the average sentiment of all the generated stories which would make the conclusions less valid.

It is important to note that schema correctness and story coherence assessment was done by one person. Even though the kappa for the agreement between the first author and the experts from Allaart's research was higher than 63%, the possibility exists that the manual assessment done by another researcher led to different results. Story coherence assessment is dependent on the English proficiency of the rater which varies from person to person and therefore could also lead to different results when done by another person. During this research the classification of schema-based stories remained an open research topic, that is why a classifier with an accuracy above change was not available. The existence of such a classifier would have made it possible to assess more generated stories and better gauge the quality of the generated stories.

7 Conclusions

A total of three candidate models were considered to generate schema-based stories namely: Generative Adversarial Networks (GAN), Recurrent Neural Networks (RNN) and OpenAI GPT. It was found that for small data sets OpenAI GPT has the best use case as RNN and GAN need data to learn grammatical structures from scratch which would require a data set with upwards of 100.000 samples. OpenAI GPT-2 does not require this as the model has already been pre-trained by its makers on a large corpus of web pages. OpenAI GPT-2 shows varying results when fine-tuned on Allaart's data set. According to the BLEU scores, the generated stories showed a maximum of 0.16 similarity with the original data set. As the n-gram increases, the BLEU score drops close to 0 which shows that as the number of words used for comparing increases the actual similarity decreases. This means that as the model is able to generate longer sentences the similarity between the original data set and the generated samples drops. Stories generated using various conditional prefixes show better consistency and schema correctness compared to the unconditional generation. A minimum of 58.7% of the conditionally generated stories was coherently and correctly classified using manual labeling while unconditionally generated stories had a maximum of 10.16%. For an actual use case, it is recommended to make separate data sets for each schema and a set of prefixes. The model should first be fine-tuned on any data set and then instructed to conditionally

generate stories for that specific schema. Based on this the OpenAI GPT-2 showed good potential to be used for actual training or verifying of binary machine learning models.

8 Future Work

This research was a showcase of what is possible using OpenAI GPT-2 for schema therapy stories. As mentioned in the limitations not all the OpenAI GPT-2 parameter versions were analyzed. Further research could delve deeper into the differences of each version and how the quality of the generated stories changes. As of March 2021, OpenAI introduced a new iteration for its GPT model namely OpenAI GPT-3 which promises to improve further on the NLP performance of its predecessor. Further research could also incorporate this new model to analyze if further improvement of the coherence and correctness can be achieved.

Traditional methods of schema therapy use SMI questionnaires to gauge the emotional state of patients over longer periods of time. It is common for human emotions to fluctuate on a daily basis as such patients would tell different stories throughout their treatment duration. This allows for patients to be classified with more than one schema at any given time. Future research could revolve around collecting stories over long periods of time and seeing how OpenAI GPT is able to mimic a variety of emotional states from each patient. This would greatly improve the usability of the chat bot as the predictions would get better as it gets to know more about the patient.

This research provides a data set with only generated stories that are manually labeled and could be used to train a chat bot. An interesting question is how a chat bot would perform solely trained on generated data as opposed to stories from real patients. Future research could shed light on what characteristics an optimal data set should have and how this affects the accuracy of the chat bot.

References

[1] David Allaart. *Schema mode assessment through a conversational agent*, 2020.

[2] Celikyilmaz Asli, Elizabeth Clark, and Jianfeng Gao. *Evaluation of Text Generation: A Survey*, 2020.

[3] Jahson Binda. *Active learning in reducing human labelling for automatic psychological text classification*, 2021.

[4] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. 2009.

[5] Akshay Budhkar, Krishnapriya Vishnubhotla, Safwan Hossain, and Frank Rudzicz. *Generative Adversarial Networks for text using word2vec intermediaries*, 2018.

[6] Franziska Burger. *Natural language processing for cognitive therapy: extracting schemas from thought records*, 2021.

[7] Dan Chisholm, Kim Sweeny, Peter Sheehan, Bruce Rasmussen, Filip Smit, Pim Cuijpers, and Shekhar Saxena. *Scaling-up treatment of depression and anxiety: a global return on investment analysis*, 2016.

[8] COCO consortium. Coco dataset, 2020.

[9] Jane Costello. *Early Detection and Prevention of Mental Health Problems: Developmental Epidemiology and Systems of Support*, 2016.

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative Adversarial Nets*, 2014.

[11] Migual Grinberg. The ultimate guide to openai's gpt-3 language model. 2020.

[12] Budi Han. *Automatic Psychological Text Analysis using k-Nearest Neighbours*, 2021.

[13] H Zar Jerrold. *Biostatistical analysis*. 1999.

[14] Kathy Kincade. Cosmogan: Training a neural network to study dark matter. 2019.

[15] OpenAI. gpt-2, 2018.

[16] Jeongwoo Park. *Automatic Psychological Text Analysis using Support Vector Machine Classification*, 2021.

[17] Yuanbin Qu, Peihan Liu, Wei Song, Lizhen Liu, and Miaomiao Cheng. A text generation and prediction system: Pre-training on new corpora using bert and gpt-2. In *2020 IEEE 10th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pages 323–326, 2020.

[18] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. *Improving Language Understanding by Generative Pre-Training*, 2018.

[19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. *Language Models are Unsupervised Multitask Learners*, 2019.

[20] Sivasurya Santhanam. *Context based Text-generation using LSTM networks*, 2020.

[21] Max Woolf. How to make custom ai-generated text with gpt-2. 2019.

[22] Tang Xiaoou, Qiao Yu, Loy Chen Change, Dong Chao, Liu Yihao, Gu Jinjin, Wu Shixiang, Yu Ke, and Wang Xintao. *ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks*, 2018.

[23] J. E. Young, J. S. Klosko, and Weishaar M. E. *Schema therapy: A practitioner's guide*. 2003.

[24] Mirijam Zhang. *Automatic Psychological Text Analysis using Recurrent Neural Networks*, 2021.

A
Intent list for schema's

intent: happy_child

- I feel loved
- I feel accepted
- I am accepted
- I am satisfied
- I feel calm

- I feel connected to other people
- Belonging
- I have stability in my life
- I have certainty in my life
- I trust people
- I feel safe
- I feel heard
- I am understood
- I am supported
- I am optimistic
- I am spontaneous

intent: vulnerable_child

- I feel worthless
- I feel inadequate
- I am not enough
- I am lost
- I feel lost
- I am desperate
- Desperation
- I am lonely
- loneliness
- I feel humiliated
- humiliation
- I feel weak
- I am helpless
- I am alone
- I feel left out
- Nobody loves me
- Nobody likes me
- I am inadequate
- I am broken
- I am excluded
- I feel powerless
- It is never enough
- I am not good
- I am a mess
- Pathetic
- My future is bleak
- I have no future
- Rejection
- I need help
- I am ashamed of myself
- I'm scared
- Fear
- I am needy
- Needy
- I am overwhelmed
- I am nervous

intent: angry_child

- I have to fight
- I am angry
- I am furious at someone
- I hold on to my anger
- I am furious
- People are with me or against me
- I am angry at someone because they left me
- It makes me angry when someone tells me wha tto do
- I want to punish people for how they treated me
- I feel cheated
- I feel treated unfairly
- I want to hurt someone for what they did tome
- I want to fight
- People are trying to limit me
- I have a lot of anger inside me
- I have to let my anger go

intent: impulsive_child

- I have trouble controlling myself
- I act first and think later
- I cannot control my impulses
- I follow my feelings
- I follow my emotions
- I get in trouble because of impulsiveness
- I do not think of consequences
- I say what I feel
- I do things impulsively
- I do first and act later
- I dont think about my actions
- I hurt people by not thinking about what I do
- I do not think
- I regret breaking rules
- I just do- Without thinking
- I did not think

intent: detached_protector

- I feel flat
- I do not feel anything
- I do not feel connected
- I do not feel my emotions
- I feel nothing

- I dont care about anything
- Nothing matters to me
- I feel distant from other people
- I feel cold
- I feel emotionless
- I do not feel connected to other people
- I do not feel connected to myself
- I am indifferent
- I dont want to feel
- I don’t like to feel
- I don’t want to
- It is not necessary
- I don’t think it helps
- It doesn’t matter
- I don’t need it

intent: punishing_parent

- I do not deserve fun
- I do not deserve enjoyment
- I do not deserve pleasure
- I do not deserve a break
- I punish myself
- Selfharm
- I injure myself
- I am a terrible person
- I am a bad friend
- I am not a good child
- I am an awful parent
- I do not forgive myself
- I am angry at myself
- I dont deserve sympathy
- I do not deserve pity
- I deserve to be punished
- It is my fault
- Bad things are my fault
- I am the cause of my problems
- I am bad
- It is my fault
- I am unsuccessful
- I can’t do it anyway
- I should be able to do this
- There is no point
- Disappointing
- I am a disappointment
- I am useless

intent: healthy_adult

- - I can solve my own problems
- I know how to express my emotions
- I can learn
- I can grow
- I can change
- I can stand up for myself
- I can assert what I need
- I know who I am
- I know what I need to be happy
- I can make myself happy
- I am a good person
- I can take care of myself
- I can handle my emotions
- I can handle bad situations
- I can do boring things
- I am happy with myself
- I am proud of myself
- My emotions do not overwhelm me
- I am stable
- I am worth the effort
- I am worth attention

B Notable samples

Happy story example 1

i had a wonderful day today because i got to know her very well. her mom and dad had a lot to talk to me about earlier today because they both love to-watch films and listen to music. they both laughed it off the experience because it wasn't something they

Happy story example 2

i had a wonderful day today because the doctors said the vaccine was working and i was doing well. however, a few hours after i had taken my last pill, she told me that she had found a lump in her breast and it was not because of the flu. she is a

Happy story example 3

i had a wonderful day today because my husband got me a job that i valued very much. this was a relief for me, as i had been struggling with depression and anxiety for years. i felt accepted and valued. i became more comfortable chatting to the person i wanted

Angry story example 1

i feel treated unfairly because of his skin colour. i don't think i can agree more. i feel sick and tired of hearing about his feelings. we have been together for 2 years now so we have obviously got our hearts set on a new relationship partner, but the issues

Angry story example 2

i have to fight because my dad never got the chance to meet me because he was too young to marry me. my family otherwise was accepting of his invitation and he would have been a really good boy. but he never showed interest or concern for his well being. he thought

Angry story example 3

i have to fight because my family doesn't want to see me die. i started a new job this week after a four year break. after i completed my eight week break, i was given another opportunity to resume. in order to do this job, i applied for