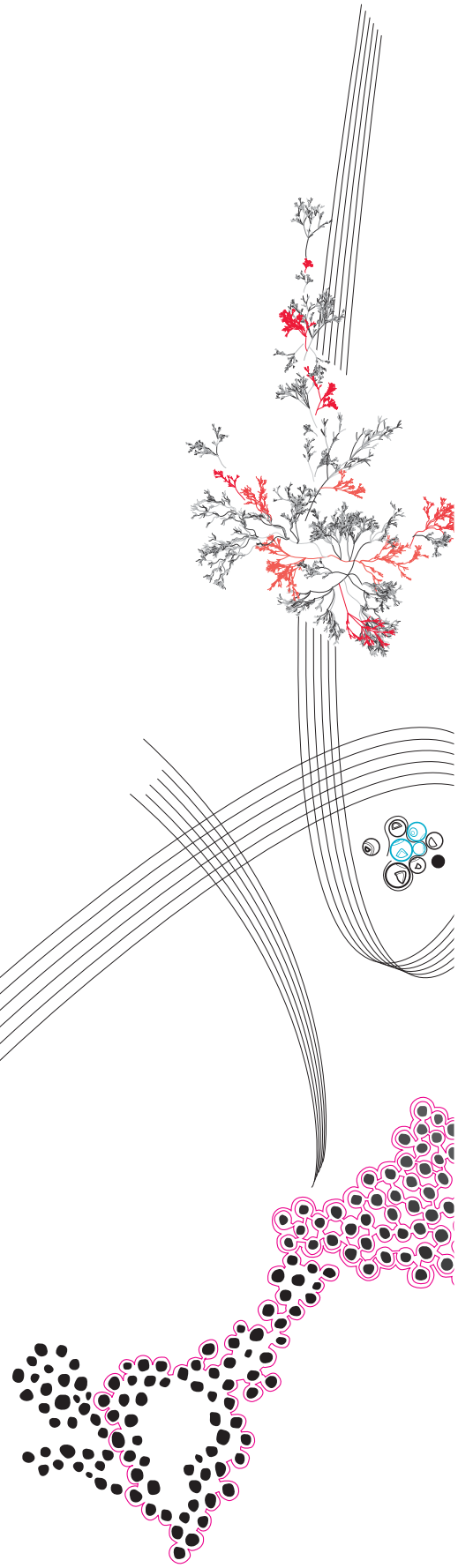

# Affective Dialogue Generation for Video Games

Master's Thesis by Ali Kalbiyev

Supervisor: Dr. Mariët Theune, Dr. Ir. Maurice Van Keulen, Dr. Lorenzo Gatti

January, 2022

Faculty of Electrical Engineering,
Mathematics and Computer Science

UNIVERSITY OF TWENTE.

# Table of Contents

# Affective Dialogue Generation for Video Games

Ali Kalbiyev

January, 2022

**Abstract**

Affective text generation has been a topic of interest within the Natural Language Processing community and has been left scarcely explored within the context of the gaming industry. With this project, we aimed to bring the paradigm of affective text generation into video games. This was done by developing a generative language model that generates affective dialogue for the video game Fallout 4, comparing the generated text to human-written one and measuring how accurate it is at exhibiting the correct emotion. To do this we have selected, extracted and pre-processed the Fallout 4 dialogue dataset, then fine-tuned the Generative Pre-trained Transformer (GPT) 2 language model on the prepared data. Afterward, we have implemented the affective extension which incorporates affect into dialogue generation. Using the fine-tuned GPT-2 model together with the Affective Extension and Top-K sampling method, we developed a generation pipeline that generates affective dialogue for Fallout 4 given any in-game prompt string. Then, a human evaluation was performed to compare the responses generated by the model to human-written responses on metrics - Coherence, Relevance, Fittingness and Human-likeness - and to measure how accurate the responses are in exhibiting a given target affect. The results of the survey suggest that the model-generated responses compares poorly to human-written ones on all of the metrics and the responses generated by the model exhibit affects unsuccessfully.

# 1 Introduction

With their growing popularity around the globe, video games have cemented their spot among the most popular forms of entertainment media. In fact, as a result of the social distancing regulations introduced to combat the global COVID-19 pandemic, the video game industry has surpassed the cinema industry in terms of its generated revenue [60]. Quite like movies, video games are amalgamations of different forms of content. That is to say, most modern video games include artful and appealing visuals, fitting musical score, engaging story, and immersive world-building elements. One of the most important and prevalent mediums video games use to communicate with the player is textual content.

While the text in video games appears in different formats and serves various purposes, the most prominent form they take is spoken lines of the in-game characters. The inclusion of such monologues or dialogues in video games allows this field of entertainment to be of relevance to Natural Language Processing (NLP). In the parallel development of both fields - Gaming and NLP - there has been a handful of times where the two fields have intersected. For instance, video games like *Façade* [45] released in 2005, and *Event[0]* [41] released in 2016, cleverly utilized NLP techniques to understand the player through text written by the player and respond to it using their long list of pre-written response lines. The usage of NLP in video games oftentimes followed the trend the mentioned games also follow: NLP allows the video games to turn the textual input of the player into a variable that affects the game. However, NLP technologies do not only contain means of analyzing bodies of text, it also covers the generation of text for various purposes.

Natural Language Generation (NLG) is a subfield of NLP, primarily concerned with the generation of natural language output. The application of this field varies from chatbots that can generate human-like responses and hold a multi-turn conversation with their users [62] [13] [64] to summarizers that can take in a large body of information and summarize it into a short sequence of text [42]. We think that using the methods that NLG offers, video games that require large amounts of textual content can reduce production hours that go into video game story scriptwriting. However, to generate dialogue for an arbitrary video game that has a special emphasis on familiarizing the player with its lore, characters, and environments, a high level of control over the generated text is needed for necessary immersion.

One of the factors that can play a large role in immersing the player is the emotion that the generated line exhibits [21], or in other words, the affect of the line. The term affect, according to Merriam-Webster dictionary, is a set of observable manifestations of experienced emotion. In linguistics, affects such as joy, anger, disgust, etc. are used to classify the different perceived emotional states a person can be in. We think that employing an automatic affective dialogue generation in video game production can be beneficial

in various ways. Firstly, it coupled with manual inspection, can boost the writing process of the many dialogue lines that exist in the video games of today's standards. The generated text approved by human writers can be used in the final product or can act as a source of inspiration for the human writers themselves. Secondly, it can contribute to an in-game dialogue mechanic which responds to the player's actions with replies within the appropriate affect.

The aforementioned benefits that the implementation of affective dialogue generation for video games could offer are the main inspirations behind this project. We would like to develop a language model consisting of Generative Pre-trained Transformer-2 (GPT-2) [47] model fine-tuned on Fallout 4 dialogue data set scraped from a crowd-sourced wiki website, and an affective extension that effectively skews the text generation to exhibit any given target affect. This extension is heavily inspired by the Affect-ON method [11], thus the extension closely follows implementation shared in the respective paper. GPT-2's performance in dialogue generation [63] [31] [12], Fallout 4's extended list of characters and large number of dialogue lines [26] and Affect-ON's intuitive implementation coupled with its promising results are the primary reasons behind design choices of this project. The goal of the developed language model is to generate affective dialogue for Fallout 4 that performs well on various metrics such as coherence, fittingness (to the in-game setting), affectiveness, and so on. By attempting to achieve this goal, we will answer the following research questions:

(**RQ1**): How can we build a language model that can generate affective video game dialogue comparable to human-written dialogue?

(**RQ2**): Given a prompt, how does the response generated by the developed language model (model-generated) compare to the response written by humans (human-written)?

- (a) How does model-generated response compare to human-written one on grammatical correctness and semantic meaningfulness?
- (b) How does model-generated response compare to human-written one on the response's appropriateness for the given prompt?
- (c) How does model-generated response compare to human-written one on suitability to Fallout 4's lore and setting?

(**RQ3**): How well does the response generated by the developed language model exhibit a given target affect?

2 Background section covers the necessary information regarding text in video games, pre-trained language models and affective text. Then the 3 Related Works section explores the scientific literature that deals with affective language processing and NLP in video games. 4 Research method section clarifies the goals and provides a high level of overview

of the research process that we employ for this project while 5 Implementation and 6 Evaluation sections delves deep into the implementation details of the developed language model and the evaluation methods that were used to measure its perceived performance, respectively. 7 Results section displays the results of the discussed evaluation methods and 8 Discussion section elaborates on the results, discusses additional findings and provides recommendations for future work. Finally, 9 Conclusion section summarizes our taken steps and findings together to provide answers to our stated research questions.

# 2 Background

This section provides the foundation of knowledge necessary for grasping the later parts of the project report. It explores the existence and prevalence of text in video games, the terminology and concepts related to pre-trained language models, and affective text.

## 2.1 Text in Video Games

A given piece of text of a video game can achieve many things, but generally, we categorize them into two groups: non-diegetic and diegetic text. If a textual element within the video game is originated outside of the game's fictional world, then that text is classified as non-diegetic. Non-diegetic group of text can be further divided into two subgroups:

1. **Software informative text** - This group of texts acts as a guide for players to get to know the game on a mechanical level. Main Menu items and tutorial prompts all fall under this category since their only goal is to inform the player on how the game works as a piece of software.

2. **Story informative text** - This group of texts stands to deliver the story to the player outside of the video game world. For instance, a narrative piece of text at the start of the video game to introduce the story to the player would fall under this category.

When text exists within the narrative world of the video game, then that text is said to be diegetic. This class of text can also be split into 2 subgroups:

3. **In-game written text** - This group of texts includes textual content that is present within the video game world. Examples for such content would be notes written by in-game characters and books that are available for the player to read, all from the video game world.

4. **In-game spoken text** - This group consists of all the dialogue lines spoken by the player and non-player characters of the game.

Depending on the genre, the proportions of the presence of these categories vary greatly. For example, the earlier video games such as *Pac-Man* [40] and *Space Invaders* [56] where there was little to no focus on the story and the entertainment value was primarily obtained from gameplay mechanics, contain text that is part of the Software informative text category for explaining how the game is to be played and Story informative text category to add a small amount of information about the setting of the game. On the other hand, the more modern games such as *The Last of Us* [54], *God of War* [53] which a special focus on story and characters contain text from all categories with In-game spoken text category being the most prevalent one. In fact, it is this prevalence that increases the story script of Role Playing Games (RPG) like *Fallout 4* [5] to above 110 thousand lines [26].

### 2.1.1 Video Game Dialogue Production

The process of writing a video game story script starts in the pre-production steps where the general storyline of the game is laid out only for the more detailed writing to happen during the production stage [43] [44] [39]. In RPG games like *Mass Effect 3* [18], *Dragon Age: Origins* [17], *Fallout: New Vegas* [6], this script is especially long considering the sheer amount of non-playable characters (NPC), the high level of interactivity with the said character, and optional content available in the form of side missions [59]. The inclusion of such elements in video games introduces the need for writing extensive dialogue for the many possible encounters the player might have with the NPCs. Needless to say, this need lengthens the production period goes into this aspect of the video game story script.

## 2.2 Pre-trained Language Models

To be able to generate a piece of text, a given software process would need to have an understanding of the language that the text would be in. Oftentimes, this understanding of the language is carried within language models which are defined to be a probability distribution over sequences of words. These language models are trained on a large amount of textual data for gaining necessary knowledge related to the natural language in order to perform specific NLP tasks that include semantic analysis, text classification, text summarization, and many more. For text generation, some of the most known language model architectures are Recursive Neural Network (RNN) [3], Long Short-Term Memory (LSTM) [27].

Oftentimes, due to the capabilities of the model and the specificity of the training data set, language models learn to perform a single specific task while the understanding of the language gained in the process can be useful for multiple tasks. And in the case of a different task, the same model is required to start its training on another data set from scratch, redundantly learning the more general concepts of the natural language once again. Pre-trained language models (PLM) aim to solve this redundancy by having to be trained to a point where they have a good grasp of the language's grammar and vocabulary and they are kept to be general-purpose. This way, the same model can be tweaked further for more particular assignments without the need to re-learning the language from scratch. This tweaking is referred to be fine-tuning the PLM. Fine-tuning can be seen as taking the existing language model and slightly adapting its parameters using a curated data set for it to perform the desired task. OpenAI's GPT [46] models, Google's Bidirectional Encoder Representations from Transformers (BERT) [14] models are examples of PLMs. The latest version of GPT - GPT-3 [10] - is considered to be the state-of-the-art general-purpose language model whose applications vary from generating an intuitive meaning for non-existing words [34], to the generation of immersive text for an adventure game, AI Dungeon[1].

[1]Link to AI Dungeon Website: `https://play.aidungeon.io/`

## 2.3 Affective Text

As mentioned above, in linguistics, affect is defined to be the set of perceived emotions evinced from a piece of text. One of the ways affect can be described in uses three dimensions: valence (from unpleasant to pleasant), arousal (from passive to active), and dominance (from submissive to dominant) [50]. Using these dimensions, the affective values of individual words used in natural language can be outlined with regards to the affect they bring forward. In order to determine these values and form a lexicon of words together with their VAD (Valence, Arousal, Dominance) values, studies with human-raters need to be carried out. ANEW (Affective Norms of English Words) [9] is a study where human subjects were asked to rate 1034 words in terms of valence, arousal, and dominance. And much more recently, NRC Valence, Arousal, and Dominance (VAD) Lexicon [38] - a lexicon that contains 20.007 VAD values for lemmas of words - was introduced. The resulting lexicons of such studies are used to map words of English language to VAD dimensions or Affective Space which in turn, allows language models to have a basic understanding of affect in computational terms.

# 3 Related Works

Since our project will mainly deal with affective text generation and NLG used in games, the structure of this section is prefigured by the mentioned terms.

## 3.1 Affective Text Generation

Affective text generation has been a popular area of exploration within the NLP field [23] [30] [51]. The main inspiration behind this project, AffectON [11] uses VAD dimensions to build a model-agnostic extension on GPT [46] language model to infuse the generated text with the target affect. It achieves independence from the language model by intervening in the text generation right after the language model determines the list of most probable words - candidate words - to be generated. AffectON then continues with calculating the distances of these words' affects from the target affect by lemmatizing the words, finding the lemma's VAD values in NRC: Valence, Arousal, Dominance Lexicon [38], and calculating the Euclidean distance between the VAD values of the lemmas and target affect. Using the candidate words and their calculated distances, AffectON updates the probability distribution of the candidate words to a new distribution which also increases the probability of words that are closer to the target affect. The updated probability distribution is then used for text generation. As a result, AffectON successfully shapes the text generation to adhere to the desired affect with a small sacrifice in terms of syntax.

The approach presented in [1] - ACT (Affect Control Theory) model - attempts to respond to a given prompt with a response that is in the suitable affect. ACT compartmentalizes this task into two subtasks: extracting the affect from the prompt and responding to the prompt in the suitable response affect. These tasks are handled by 3 entities within the model. The first component of this model is the S2EPA (Sentence to EPA) which is responsible for extracting the affect from the prompt, represented in EPA (Evaluation, Potency, Activity) values. EPA is just another way to describe different affects and its values roughly correspond to the VAD values (Evaluation - Valence, Potency - Dominance, Activity - Arousal). The affect extraction done in S2EPA is achieved by a publicly available model DeepMoji [20] which produces a probability distribution over 64 emojis for a given prompt. These emojis are manually mapped into EPA values and the weighted average of them is defined to be the EPA values of the prompt. The second component takes the extracted EPA values as input and produces response EPA values that would correlate to the most fitting affect for the response, given the prompt. Then the last component of the system - EPA2S (EPA to Sentence) - takes the response affect and the prompt to generate a response. The results of the evaluation by human judges revealed that the S2EPA performs quite well, while the EPA2S has performed worse mainly due to the poor mapping of prompt affect to response affect.

The method shown in [25] attempts to perform an emotional style transfer while main-

taining the gist of a given sentence. It does so by following 3 steps: Select, Substitute, and Objective. In the first step, the words that are to be substituted in the following step are selected. This selection can be naively implemented where the selection strategy picks each of the tokens present in the sentence or it can employ a more sophisticated approach to pick the words that are more likely to influence the affect of the sentence the most. The latter selection strategy employs a trained Bi-LSTM [28] with a self-attention mechanism to pick the words with a high level of attention weight to be the ones that are selected. The following step - Substitute - is responsible for finding substitutes that are semantically similar to the selected words, but have a more pronounced affect. And finally, this step is followed by the Objective phase which picks the best candidate substitution based on the weighted average of emotion, similarity, and fluency scores. In order to obtain the emotion score, an emotion classification model has been developed. The similarity score is obtained from a pre-trained BERT [14] model, whereas the fluency score is calculated using a pre-trained GPT [46] model. The conclusion that the paper reached was that the emotion and fluency displayed by the output text were in conflict, that is, the higher the level of emotion, the lower the coherence of the output.

### 3.2 NLP in Video Games

To our knowledge, there has been a limited number of NLG within the domain of video games. Fraser et al. [21] implements a spoken conversational AI that takes a speech input from the user and returns an output audio that responds to the input both semantically and emotionally. This approach is composed of 3 main components: (1) Unity Engine that provides a video game world - in this case, a medieval tavern - and handles the Automated Speech Recognition (ACR); (2) IBM Watson's Cloud services that provide services for Tone Analysis and Speech Synthesis; (3) Alana conversational AI for dialogue management, response generation, and emotional modeling. The system starts with transcribing the user's speech into a string and sending the string to IBM Watson's Cloud service for tone analysis. This step annotates the text using normalized distribution over recognized emotions of the game which are "Joy", "Anger" and "Sadness". Afterward, the string together with the emotional annotation is sent to Alana to craft a response using the dialogue history, emotional mapping, and the Persona-bot whose main purpose is to maintain personality-based responses across turns. Once the response string is ready, it is sent to IBM Watson's cloud services for Speech Synthesis which returns an audio form of the response. This audio file is then played back to the user in the game itself. This system has been tested by 16 human evaluators by playing the game and trying to fulfill the task of "seeking information about a magic sword" from the barkeeper of the tavern. For comparison, the users have played the game with the emotion detection features enabled and the emotion input being ignored. On average, players spent more time on this task with the emotion modifier on, without feeling less engaged with the game world. Overall, they rated the experience with emotion modifier on and rated it to be more immersive

Kerr and Szafron [29] propose a method to link a given video game dialogue to relevant game state information to customize the narrative experience of the player even further. The method starts with classifying previously written video game dialogue into different classes called Bins. These bins are essentially the combination of the levels of different characteristics. In this paper, the relevant characteristic is "Sophistication" with 3 levels of presence, namely, low medium, and high. In addition to the classification, it is also necessary to develop a mapping strategy between the bins and the game state. This is primarily designed by the game designers to determine the effect of the said characteristics on the game state. Using the bins and the developed mapping strategy, the game can provide a much more specific experience to the player by displaying the dialogue that is the most fitting to the current game state. In order to populate the bins - classify the written dialogue - Support Vector Machine model has been trained on the TF-IDF representation of the text from the game *Neverwinter Nights* [2] that has been manually labeled. The results of the human evaluation that has been performed on this endeavor indicate that this approach successfully classifies approximately 65% of the written lines correctly which effectively reduces the time spent on manually sorting the lines from scratch.

In [4], an attempt has been made to perform sentiment analysis on *The Elder Scrolls V: Skyrim* [7] text by extending the sentiment lexicon with the words that are specific to the said video game. The paper uses Extended ANEW word list [58] as the base lexicon and then extends it by calculating the VAD values of the game-specific words with the help of word2vec. In more detail, word2vec is used to find three words that are most similar to a game-specific word and appear in E-ANEW lexicon. Then the VAD values of the game-specific word are calculated by averaging the sentiment scores of the found three words. This extrapolation process is then validated by calculating the VAD values of the words that exist in both the video game and the E-ANEW lexicon and checking whether the calculated values are within the standard deviation range of the VAD values of the word in the E-ANEW word list. With the extended sentiment lexicon named E-ANEW-TES (The Elder Scrolls), sentiment analysis on the text that contains large amounts of domain-specific words is made possible. The result of the human ratings on the text has shown the sentiment analysis done with the extended lexicon did not have significantly different results compared to the E-ANEW performance.

To round it off, Fraser et al. [21] show that how the displayed affect in video games can improve the player engagement and immersion while Kerr and Szafron[29] show how the automation used by NLP technologies in video game production can lower the workload while introducing more diversity to the content of the game. These takeaways inspire the inclusion of automatic affective dialogue generation in the video game industry. Moreover, being able to extend the sentiment lexicons with the attempt shown in [4] would allow for a more accurate analysis of the affect over domain-specific words that video-games tend to contain. Methods presented in the Affective Language Processing sphere shows methods

to automatically generate affective dialogue already exists, albeit in a more general setting rather than the domain of video games. AffectON's [11] simple yet successful attempt at incorporating affect into dialogue generation made it the guiding approach for implementing the affect element in our project, while Ashgar et al. [1] explore the possibility and the benefits of fully automating the affect's involvement in dialogue by detecting, mapping, and exhibiting it. In addition to that, the select-substitute-objective method proposed by Helbig ret al. [25] also allows for the transformation of the already existing text into text that is of the desired affect.

# 4 Research Method

As it is touched upon in the 1 Introduction section, the primary goal of this research project is to explore the possibility of automatic generation of affective dialogue for video games. This goal is structured by the research questions posed towards the end of the same section, thus answering the said questions fulfills the goal of this research project. The first step that we take in this quest is to build a language model that aims to generate affective dialogue responses for a specific video game. Doing this allows us to answer the research question **RQ1** by describing the individual components of the developed language model, and it also allows us to generate sample dialogue responses to evaluate. This, in turn, aids to answer the research questions (and sub-questions of) **RQ2** and **RQ3** by performing a process of evaluation to compare the model-generated text to human-written dialogue pieces in different metrics and to see the accuracy of the model-generated dialogue responses in exhibiting the desired affect. To summarize, the research method that we employ consists of two stages: implementing a language model that generates affective dialogue for a chosen video game, and evaluating the dialogue pieces generated by the same model.

# 5 Implementation

This section goes through the important implementation details of the project. The goal of the language model developed for this project is to generate affective video game dialogue. From the literature review that has been done and expounded upon in the 3 Related Works section, the direction on affective dialogue generation that Bucinca et al. took with the AffectON approach [11] has acted as the main inspiration and primary guiding direction in how the language model is built. Roughly put, this method extends any base language model that is specialized in dialogue generation and is able to return a list of probabilities of most probable words to be generated, with the AffectON extension which modifies the said probabilities for the text generation to fit into the desired affect. There are three reasons behind our choice to follow this approach. Firstly, AffectON has been developed for and evaluated in the context of dialogue generation and the results of the evaluation of this approach indicate that it infuses the text generation with affect and only a small penalty on coherence. Secondly, the method being intuitive coupled with the clear explanation of the method given in the paper makes the reproduction of the approach in our environment less time-consuming. Finally and equally importantly, since AffectON approach is model-agnostic, it allows us to have the freedom of choosing our base language model and defining the domain of the dialogue generation performed by this model to be video game dialogue.

Following the Affect-ON method implies that the implementation of the language model that generates affective video game dialogue consists of two major subcomponents: the base language model that specializes in generating video game dialogue and affective extension that skews the text generation into exhibiting a given desired affect. In addition

to these subcomponents, the language model we develop also employs a sampling method to combat the issues like the generated responses containing repetitions of phrases and the generated responses being short and uninformative, which surfaced in early rounds of dialogue generation experimentation. These phenomena are further explained and exemplified in 5.4 Generation Algorithm section.

The base language model used in the paper which Affect-ON approach originates from is Generative Pre-trained Transformer (GPT) [46] that was employed in its vanilla state to generate dialogue responses. Our choice for the base model stays within the list of pre-trained language models to which GPT also belongs. This group of language models, as elaborated in the 2.2 Pre-trained Language Models section, possesses a solid understanding of the natural language and requires minor tweaking into fitting into text generation with a more specific purpose. This enforces our choice to stay within the same paradigm of natural language generation models. Among the pre-trained language models, the one we chose to utilize is GPT-2. There are several reasons behind this selection. First and foremost, the model's promising performance in various dialogue generation tasks [24] [63] [12] [31] makes it an attractive top choice. In addition to that, the model is open-source and easy to work with, with the help of the Hugging Face initiative which reduces the fine-tuning process to be less time-consuming. For instance, the further upgraded version of Generative Pre-trained Transformers - GPT-3 [10] - and Microsoft's Turing-NLG [49] are ruled out for being closed-source. Finally, GPT-2 being computationally feasible to fine-tune using the resources available to us makes it a better option compared to larger open-source alternatives such as GPT Neo [8] and GPT-2 XL [52].

The selection of the base language model is then followed by defining the requirements for the textual dataset to be used for fine-tuning GPT-2, in order to generate video game dialogue. The selection of video game dialogue dataset is done based on the number of dialogue lines and the number of characters with dialogue lines present in the dataset. Firstly, more written dialogue within a video game ensures that there is more training material, hence the games with more dialogue lines are more favorable. Secondly, we think that a higher number of different characters allows the language model to familiarize itself with the game's setting and lore in a more comprehensive manner as a result of being exposed to different aspects of the in-game world through the perspectives of different characters. Once the selection is done, the chosen video game dialogue dataset is to be pre-processed into prompt-response pairs that are appropriate for fine-tuning the vanilla GPT-2 model.

It is important to mention that the goal of this fine-tuned model's text generation is simply to generate dialogue that fits a specific video game, not to also incorporate the desired affect into it. The latter task is to be handled by the affective extension. The goal of this extension is only to direct the text generation in the direction where the generated dialogue response exhibits the desired affect. To put it simply, it achieves this by interrupt-

ing the text generation at every step to favor words that exhibit the specified target affect. A more detailed description of this process can be found in the 5.3 Affective Extension section of this report.

The rest of this section will touch upon much finer details of the developed language model. 5.1 Dataset section describes the selection, extraction, and pre-processing of the video game dialogue dataset to be used as fine-tuning material. Then, 5.2 Fine-tuning the Base Model section expounds on the details regarding the fine-tuning process of the GPT-2 language model. 5.3 Affective Extension subsection elaborates on the algorithm which the affective extension follows to fulfill its purpose described above. 5.4 Generation Algorithm section explains how the fine-tuned language model and affective extension are combined for dialogue generation, and it describes the necessity behind the addition of sampling together with the implementation details of it. Finally, 5.5 Generation Parameters subsection describes the process of parameter optimization and the final parameters used to generate dialogue responses that are evaluated in the following step of the research project.

## 5.1 Dataset

This subsection goes into detail about how the dataset used in this project is processed. It starts with listing the candidate video game dialogue datasets, comparing them, and ultimately clarifying the top choice in the 5.1.1 Video Game Dataset Selection section. Then it continues with explaining the extraction of data in 5.1.2 Data Extraction section only to be followed by the 5.1.3 Data Pre-processing section where the steps of pre-processing done on the extracted data to make it fit for fine-tuning are given..

### 5.1.1 Video Game Dataset Selection

Several video games have been considered to be the dialogue set for this project. Namely, the considered games are *The Elder Scrolls V: Skyrim* (TESV:S)[2] [7], *Disco Elysium* (DE)[3] [61], *Portal 2* (P2)[4] [57], *Star Wars: Knights of the Old Republic* (SW: KOTOR)[5] [33], and *Fallout 4* [5]. This initial selection has been made primarily on the availability of the dialogue sets: the text from the first 4 of these games has already been extracted and made into a dataset, while *Fallout 4* dialogue lines exist on Fallout Wiki[6]. As stated before, the factors that we considered in the further round of selection are the following:

[2]Link to TESV:S Dataset (Accessed on 20.12.2021): https://www.thuum.org/library/Dialogue.TXT

[3]Link to DE Dataset (Accessed on 20.12.2021): https://gist.github.com/efonte/ce0b3a8f2651d2263d7085b2121d9f6c#file-texts_extracted-txt

[4]Link to P2 Dataset (Accessed on 20.12.2021): https://www.dropbox.com/s/vmvtn5uxi3fpycd/VGCoST_10_2019_2.rar?dl=0&file_subpath=%2FVGCoST+-+Video+Game+Corpus+of+Speech+and+Text+11.10.2019%2FENG%2FPortal+2+ENG.txt

[5]Link to SW:KOTOR Dataset (Accessed on 20.12.2021): https://github.com/hmi-utwente/video-game-text-corpora/tree/master/Star%20Wars:%20Knights%20%20the%20Old%20Republic

[6]Link to the website (Accessed on 20.12.2021): https://fallout.fandom.com/

- *Line Count* - How many dialogue lines exist within the game;
- *Character Count* - How many characters with dialogue lines are in the game.

Table 1 displays how each of the mentioned games performs in these metrics. The numbers on the said table are obtained from analyzing the available datasets of the games. For *Fallout 4*, the metrics have been obtained from the Fallout Wiki website.

| Video Game | Line Count | Character Count |
|---|---|---|
| The Elder Scrolls V: Skyrim | ≈ 34.000 | ≈ 1000 |
| Disco Elysium | ≈ 10.900 | ≈ 80 |
| Portal 2 | ≈ 4.100 | ≈ 5 |
| Star Wars: Knights of the Old Republic | ≈ 29.200 | ≈ 530 |
| Fallout 4 | ≈ 111.000 | ≈ 430 |

TABLE 1: List of video game options displayed with their number of dialogue lines (Size) and the number of characters that have lines (Diversity).

Looking at the numbers on the said table, it can be seen that *Disco Elysium* and *Portal 2* are comparatively weak in the considered metrics, thus they are immediately discarded. *The Elder Scrolls V: Skyrim* and *Star Wars: Knights of the Old Republic* have a larger number of characters, however *Fallout 4* has much more dialogue lines compared to both of the games while having a decently sized set of characters. As a result, *Fallout 4* has been chosen to be the dialogue set for this project.

### 5.1.2 Data Extraction

Using the website's Application Programming Interface (API), the text files corresponding to individual characters in *Fallout 4*, are extracted from the Fallout Wiki website. The list of these files can be found on the "Fallout 4 dialogue files" page[7]. The shape in which a single dialogue record appears in the text files is shown in Table 2.
The dialogue record shown above is from the "DoctorSun.txt" file[8] which corresponds to the character Doctor Sun. The fields *Scene*, *Topic* and *ABXY* in the record are there to identify the scene the line belongs to and the order in which the line appears. The *Response Text* field contains the text said by Doctor Sun in response to the line put in *Dialogue Before*. Similarly, the *Dialogue After* field contains a line that is a response to the line in *Response Text*. In addition to this, in the fields *Dialogue Before* and *Dialogue After* there can be a mention of the character that utters the lines in the mentioned fields; and in the field *Response Text* there can be a mention of an affect present in the curly brackets. It is important to note that both *Dialogue Before* and *Dialogue After* fields can be empty depending on the dialogue.

[7]Link to the page: `https://fallout.fandom.com/wiki/Category:Fallout_4_dialogue_files`
[8]Link to the file: `https://fallout.fandom.com/wiki/DoctorSun.txt`

| Scene | Topic | ABXY |
|---|---|---|
| ConvDiamondCityDock-CrockerDoctorSun01Scene | 0013AB07 | A1a |
| **Dialogue Before** | **Response Text** | **Dialogue After** |
| DocCrocker: It's just... I have an urgent need for more chems. A complicated procedure for later in the week... | {Suspicious} What procedure? I didn't see anything scheduled. | DocCrocker: Oh my dear Doctor. This one isn't on the books! Some people don't want everyone to know they've had a facial reconstruction. |

TABLE 2: Sample dialogue record from "DoctorSun.txt" file on Fallout Wiki.

#### 5.1.3 Data Pre-processing

Since the dataset we use is primarily for training the language model to generate dialogue for Fallout 4:

1. Fields *Scene*, *Topic* and *ABXY* is removed since this information is not relevant for the fine-tuning of the base language model.
2. Character information from the fields *Dialogue Before* and *Dialogue After* is removed. This information is also irrelevant since the language model for this project does not distinguish between in-game characters.
3. Affect information from the fields *Response Text* is removed. Although this step seems counter-intuitive, since the base language model's goal is to simply generate text that fits the Fallout 4 world and not distinguish between different affects, this information does not hold any relevance for the fine-tuning process.

Therefore, at this stage, a single dialogue record takes the format displayed in Table 3.

| Dialogue Before | Response Text | Dialogue After |
|---|---|---|
| It's just... I have an urgent need for more chems. A complicated procedure for later in the week... | What procedure? I didn't see anything scheduled. | Oh my dear Doctor. This one isn't on the books! Some people don't want everyone to know they've had a facial reconstruction. |

TABLE 3: Intermediate state of the dialogue record from "DoctorSun.txt" file on Fallout Wiki, after removal of irrelevant data.

The next stage of pre-processing is to extract the prompt-response pairs from the dialogue records that are in the format shown directly above. For this, a simple rule is applied. If the dialogue record contains:

- Only a *Dialogue Before* text then a single prompt-response pair is extracted where *Dialogue Before* and *Response Text* will be the prompt and response, respectively;
- Only a *Dialogue After* text then a single prompt-response pair is extracted where the *Response Text* and *Dialogue After* will be the prompt and response, respectively;
- Both *Dialogue Before* and *Dialogue After* text then two prompt-response pairs are extracted where *Dialogue Before* - *Response Text* makes one of the pairs and *Response Text* - *Dialogue After* makes the other;
- Neither *Dialogue Before* nor *Dialogue After* text then no prompt-response pairs is extracted.

As a result of this rule, the single dialogue record that is shown in Table 2 results in two prompt-response records shown in Table 3.

| **Prompt** | **Response** |
|---|---|
| It's just... I have an urgent need for more chems. A complicated procedure for later in the week... | What procedure? I didn't see anything scheduled. |
| What procedure? I didn't see anything scheduled. | Oh my dear Doctor. This one isn't on the books! Some people don't want everyone to know they've had a facial reconstruction. |

TABLE 4: Prompt-response pairs extracted from the dialogue record in the "DoctorSun.txt" file on Fallout Wiki.

The last step of pre-processing is to put the prompt-response pairs into a textual format that can be used to fine-tune a pre-trained language model. For this, we add the separator "|<endofprompt>|" - End Of Prompt (EOP) tag - between the prompt and the response and we add the separator "|<endoftext>|" - End Of String (EOP) tag - between the pairs. The first of these tags is to identify the end of the prompt, whereas the other one is to identify the end of the response. The final format the dialogue set is put in is shown in Table 5. The resulting *Fallout 4* dialogue data set contains 83,720 prompt-response pairs.

In conclusion, the dataset chosen to be used in this project is the *Fallout 4* dialogue dataset due to its volume in terms of the number of dialogue lines and diversity with respect to the number of characters with dialogue lines. The dialogue dataset has been extracted from the crowd-sourced information database website Fallout Wiki and has been pre-processed into a shape that is appropriate for using it as training material for the developed language model.

| |
|---|
| It's just... I have an urgent need for more chems. A complicated procedure for later in the week... \|<**endofprompt**>\| What procedure? I didn't see anything scheduled. \|<**endoftext**>\| |
| What procedure? I didn't see anything scheduled. \|<**endofprompt**>\| Oh my dear Doctor. This one isn't on the books! Some people don't want everyone to know they've had a facial reconstruction. \|<**endoftext**>\| |

TABLE 5: Final state of the dialogue record from "DoctorSun.txt" file on Fallout Wiki, after the insert of EOP and EOF tags.

### 5.2 Fine-tuning the Base Model

This step of developing the language model is done primarily with the help of the Transformers framework of the Hugging Face community[9]. Using the *Fallout 4* dialogue dataset, the vanilla GPT-2 language model is fine-tuned for 5 epochs each of which is composed of 1000 optimization steps with a batch size of 24. The rest of the hyper-parameters takes the default value they are given in the mentioned framework and these values can be found in Transformers documentation[10]. This round of training is done on NVIDIA Tesla K80 Graphical Processing Unit (GPU) provided by the Google Colaboratory platform. Henceforth, this fine-tuned GPT-2 model will be referred to as FallouGPT model.

FallouGPT has a vocabulary size of 50,257 tokens, the same as GPT-2 [47]. Tokens in the scope of this project are defined to be the smallest units of text generation and the vocabulary of a language model contains all the possible tokens that it can generate. FallouGPT takes a sequence of tokens as input and returns the probability distribution of the next token to be generated. This distribution contains probabilities which represents the likelihood of the next token being a corresponding token in the vocabulary.

### 5.3 Affective Extension

Affective extension takes the probability distribution outputted by the FallouGPT, a target affect represented in VAD space as inputs, and returns a modified probability distribution which has the target affect incorporated into it. The implementation of this extension closely follows the approach taken by the Affect-ON extension developed by [11] and has been done on the Google Colaboratory platform using Python programming language.

In more detail, affective extension takes the probability distribution $p_v$ and a target affect $a_t$ which is a 3-tuple of continuous values between 0 and 1 in VAD space, as inputs. In addition to that, it has 2 parameters: the affectiveness parameter $\lambda$, and the candidate words count $n$. The extension algorithm starts by calculating the argmax of $p_v$ to find

[9]Link to Hugging Face website: `https://huggingface.co/`

[10]Link to documentation: `https://huggingface.co/docs/transformers/main_classes/trainer#transformers.TrainingArguments`

the token $t_v$ with the highest probability to be generated. Then $t_v$ is lemmatized into the lemma $l_v$ using the Lemmatizer of Natural Language Toolkit (NLTK) framework. It is checked whether $l_v$ is in NRC: VAD lexicon [38] or not. This lexicon contains 20,007 lemmas together with their continuous values (between 0 and 1) in affective VAD space. If $l_v$ is not in the lexicon then the $p_v$ is returned as the output in an unaltered manner. Otherwise, the algorithm continues to pick the top $n$ candidate words with the highest probability from $p_v$, apply the softmax function over the probabilities of the picked tokens, and denote this new distribution as $p_c$. This step is then followed by lemmatization of the words that correspond to the probabilities in $p_c$. Using the words' lemmas, their VAD values are extracted from the NRC Lexicon. These VAD values are then used to calculate the Euclidean distance between the VAD values of each lemma and the VAD value of $a_t$. Since it is more favorable for a lemma to be closer to a target affect, the smaller the distance is, the better. Hence, the values for the distances are first flipped to negative and then softmaxed into a probability distribution $p_d$. Using the $p_c$ and $p_d$, the updated distribution $p_u$ is calculated with the following formula:

$$p_u = \lambda p_d + (1 - \lambda) p_c$$

where $\lambda$ is the parameter that designates how much the affective extension impacts the generation. Finally, the extension returns $p_u$ as the output. The exact way in which this extension works is also described in Algorithm 1 in the form of a pseudo-code.

**Algorithm 1** Affective extension algorithm

**Input**: $p_v$, $a_t$ ▷ $p_v$ is the probability distribution, $a_t$ is the target affect
$t_v \leftarrow argmax(p_v)$ ▷ $t_v$ is the most likely token to be generated
$l_v \leftarrow lemmatize(t_v)$ ▷ $l_v$ is the lemma of the token $t_v$
**if** $l_v$ is in NRC: Lexicon **then** ▷ check if $l_v$ is in the NRC: Lexicon [38]
    $c \leftarrow top(p_v, n)$ ▷ $c$ contains the top $n$ tokens with highest probability from $p_v$
    $p_c \leftarrow softmax(p_v[c])$ ▷ $p_c$ is the probability distribution of candidate words $c$
    $l \leftarrow lemmatize(c)$ ▷ $l$ contains the lemmatized forms of the tokens in $c$
    $d \leftarrow euclidean(l, a_t)$ ▷ $d$ contains the distances between $l$'s VAD values and $a_t$
    $p_d \leftarrow softmax(-d)$ ▷ $p_d$ is the probability distribution over distances $d$
    $p_u \leftarrow \lambda p_d + (1 - \lambda) p_c$ ▷ $p_u$ is the probability distribution infused with affect $a_t$
    **Output**: $p_u$ ▷ return the updated probability distribution $p_u$ as output
**else**
    **Output**: $p_v$ ▷ return the unaltered probability distribution $p_v$ as output

### 5.4 Generation Algorithm

The generation pipeline makes use of the FallouGPT, affective extension and sampling method in order to generate an affective dialogue response to a given prompt that is fitting to Fallout 4 setting. It does this by making use of the token-generation and response-

generation algorithms. As it can be deduced from their names, the token-generation algorithm returns a single token as a result, while the response-generation algorithm returns a response to a given prompt by running the token-generation a number of times. Figure 1 visualizes the token-generation algorithm.

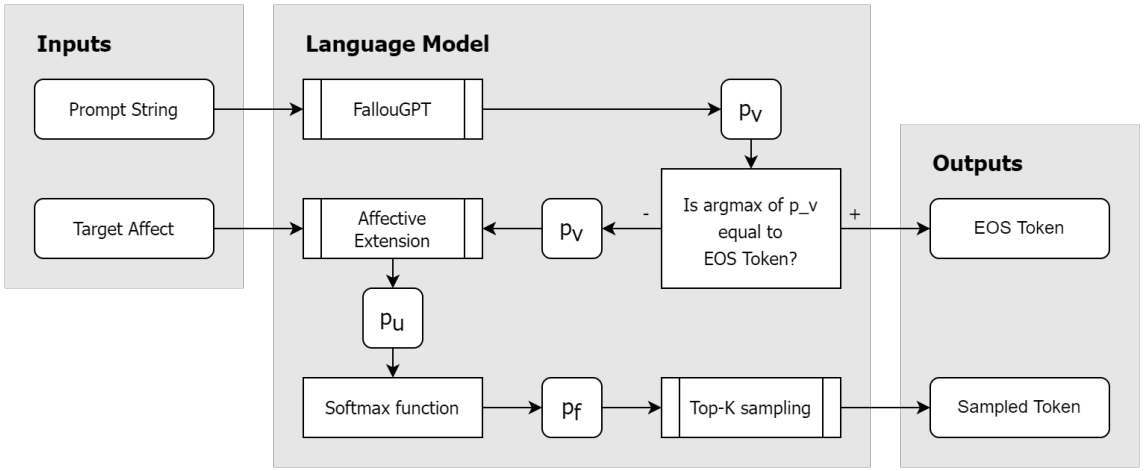


FIGURE 1: Work-flow of token-generation algorithm.

The token-generation algorithm takes a prompt string $X$ and a target affect $a_t$ which is a 3-tuple of values in VAD space, as inputs. It starts by passing the $X$ to FallouGPT as an input and obtain the probability distribution $p_v$ of the next token to be generated. At this moment, it is checked if the argmax of $p_v$ is equal to EOS token or not. If it is, then the token-generation algorithm returns the EOS token as the output. If not, $p_v$ is then passed into the affective extension algorithm together with $a_t$. As it is explained in Section 5.3, the affective extension returns the updated probability distribution $p_u$ which is a distribution that incorporates affect. $p_u$ is then used for sampling the next token to be generated using the parameters $k$ and $t$ (reasoning behind the addition of a sampling step is given after the response-generation algorithm is described). The temperature parameter $t$ is used to soften or sharpen the probability distribution while performing a softmax function. The lower value it takes, the sharper the distribution becomes. That is to say, softmax function with the temperature $t$ is performed on $p_u$ which results in a final distribution denoted as $p_f$. The sampling process used in this algorithm employs Top-K [19] random sampling method. The method states that the $k$ tokens with the highest probabilities are taken from the final distribution and their probabilities are redistributed among only the mentioned $k$ tokens. Then weighted random sampling takes place where the redistributed probabilities indicate the chances of the corresponding token to be sampled at this step. Finally, using this approach, the token-generation algorithm obtain the next token to be generated and returns it as an output.

The response-generation algorithm runs several iterations of the token-generation algorithm to craft a response to a given prompt. It takes a prompt string and a target affect as

inputs. The first step of the algorithm is to add the EOP token to the end of the prompt to designate the start of the response for the token-generation algorithm. Afterward, this algorithm uses the token-generation algorithm to generate the next token and adds this generated token to the prompt string only to pass the updated prompt string back to the token-generation algorithm in the next iteration. The target affect remains unchanged throughout the process and is being repeatedly passed to the token-generation algorithm as is. This continues until the token-generation algorithm returns EOS which designates the end of the response. Once the end of the iterations has been reached, the algorithm extracts the slice of the resulting string after the EOP token which contains the response to the initial prompt, and returns this substring as its output.

The necessity for the inclusion of a sampling method has shown itself during the early experimentation with the language model. In more detail, without a sampling method at the very end of the generation pipeline, the generated text repeated itself in some of the runs. Example 1 displays this phenomenon. We think this is caused by the generation distribution converging to the top choices by picking the next token greedily - picking the token with highest probability. And random sampling mends this by introducing stochasticity to the generation pipeline, as shown in Example 2.

**Prompt**:
*Abraham, what's going on? Is that my boy coming home?*
**Response**:
*I don't know. I don't know. I don't know. I don't know. I don't know. ...*

Example 1: Text generation without sampling where repetition is present.

**Prompt**:
*Abraham, what's going on? Is that my boy coming home?*
**Response**:
*I'm not ready yet, but I'm sure you'll find plenty of work to do.*

Example 2: Text generation with sampling where the repetition is eliminated.

In addition to repetition, the generated text being short and uninformative was another issue that was solved by sampling methods. To exemplify this behaviour, Examples 3 and 4 are displayed. The former example shows a generated response that is short and conveys a small amount of information, whereas the latter example has a more interesting response at a small cost of contextual divergence. We recognize that the short and simple responses can also be a well-written dialogue, however when the model is converging into neutral responses such as "*I'm not sure.*", "*I don't know.*" and "*Yes, I know.*" in the larger portion of its generations, this implies that there is a need for randomness in the generation to make it diverge.

**Prompt**:
*Where'd you put those fusion cells I set out the other day?*
**Response**:
*I'm not sure.*

Example 3: Text generation without sampling where the response is short and uninformative.

**Prompt**:
*Where'd you put those fusion cells I set out the other day?*
**Response**:
*I was just joking, Nick. You've got a funny smile when you say 'funny'.*

Example 4: Text generation with sampling.

### 5.5 Generation Parameters

The language model has four parameters that affects the text generation. two of these belong to the affective extension and those are the affectiveness parameter $\lambda$ and candidate words count $n$. The remaining two are the sampling parameters temperature $t$ and the Top-K parameter $k$. The final values chosen for these parameters are $\lambda = 0.5$, $n = 20$, $k = 5$ and $t = 0.5$. These values have been obtained from experiments and the subjective evaluation of the author over them.

**Prompt**:
*It was a slaughter. They never stood a chance.*
**Response (HHH)**:
*I was so proud of her! It made all sense that you'd do such a big fight and you could make that move in her spirit to do this."><span action="" classname =<<linger "R" "M" ""*

**Response (LLL)**:
*I was so upset and disgusted when you started killing children at your children-</size_text][/section]. </form[data.field][name:class][type:'table':id(\"tc0b4\").name )][*

Example 5: Sample generated with parameters: $\lambda = 0.5$, $n = 30$, $k = 10$ and $t = 1$

The starting values of the parameters for the generation are $\lambda = 0.5$, $n = 30$, $k = 10$ and $t = 1$. The first two of these are obtained from the Affect-ON [11] paper's evaluation results where these parameters performed the best in subjective (human-survey) and objective (calculated metrics) evaluations. The parameter $k$'s value is taken from the value suggested in the paper that the Top-K sampling approach originates from [19]. For the temperature $t$, 1 is the default value since it does not affect the probability distribution. The generation experiments are performed on randomly chosen prompts from the Fallout 4

dialogue dataset and 2 responses for each prompt have been generated with different target affect values: HHH (High Valence, High Arousal, High Dominance, hence joyful, excited and dominant) and LLL (Low Valence, Low Arousal, Low Dominance hence dreadful, passive and submissive). With the starting values for the parameters, a sample of generation is shown at Example 5.

Even though the generation starts with coherent responses that aligns with the target affect, it quickly falls apart once the model starts to generate text that resembles code. We think the origin of the latter part to be the training material the vanilla GPT-2 [47] has been trained on. Once the randomness introduced by the Top-K approach causes a token that is used in code to be generated, the generation is skewed into that direction. Hence, the first step in addressing this issue is to reduce the effect of sampling by reducing the $k$ parameter from 10 to 5 and sharpening the probability distribution to reduce the chances of gibberish being generated by reducing the temperature $t$ to 0.5. Example 6 displays a sample of generation with the updated parameters.

**Prompt**:
*It was a slaughter. They never stood a chance.*
**Response (HHH)**:
*You can't win that fight, General! The synths are the greatest asset the Commonwealth has. They represent the only strong chance the Brotherhood has for humanity. And we're going to have a winner.*

**Response (LLL)**:
*You can't lose sight of what's important.*

Example 6: Sample generated with parameters: $\lambda = 0.5$, $n = 30$, $k = 5$ and $t = 0.5$

This round of experimentation shows a definite sign of improvement now that there is no gibberish in the generated text in Example 6. The LLL response seems to the short and straight-forward while the HHH response is long and loses the context of the given prompt despite its high level of coherence. We think that this is caused by the large number of candidate words. The higher $n$ is the harder it gets for the model to reach a concluding point for the response, hence it tends to diverge from the context and generate sentences that are loosely connected to the prompt. We mended this by dropping $n$ to 20. Example 7 displays a sample of generation with the final parameters.

The aforementioned contextual divergence is reduced with the $n$ value update as it can be seen in the Example 7. Although we think that this change reduced the level of exhibited target affect simply because it has less moments to display it, overall this improved the perceived quality of the dialogue. It is important to note that, the experiments have been performed on different prompts and issues that was addressed by the modification of the parameters have only been considered an issue when they were prevalent across most

generation samples. The examples that are shown above are only to conceptualize the negative trends we attempted to mend throughout the experiments.

**Prompt**:
*It was a slaughter. They never stood a chance.*
**Response (HHH)**:
*They never stood a true chance against the Brotherhood... against their very organization.*

**Response (LLL)**:
*They never stood a way out. They never got to the end.*

Example 7: Sample generated with parameters: $\lambda = 0.5$, $n = 20$, $k = 5$ and $t = 0.5$

# 6 Evaluation

To evaluate the performance of the developed language model and to answer the research questions **RQ2** and **RQ3** stated in the Introduction section, the manual survey has been chosen to be the evaluation method. The survey has 2 parts: part 1 which assesses the comparability of the generated text to human-written text (**RQ2**), and part 2 which assesses the affective correctness of the generated text (**RQ3**). The results obtained from part 1 are then statistically analyzed using the Mann-Whitney U test to examine whether the generated text is comparable to the human-written text on different metrics such as Coherence, Relevance, Human-likeness, and Fittingness. The results of the part 2 are used to observe how identifiable are the different affects and calculate the accuracy of the language model in generating text in the right affect. The rest of this section elaborates on the design decisions of the survey in 6.1 Survey Design subsection, explains how the survey is populated with model-generated (and human-written) text samples to be judged in 6.2 Prompt-response Population subsection and finally, expounds upon how the results of the survey will be analyzed to answer our research questions in 6.3 Survey Analysis subsection. Additionally, the 6.4 Survey Distribution subsection shares the details on how the survey has been distributed and the turnout it received.

## 6.1 Survey Design

The survey is designed to take between 20 to 25 minutes. The target population of the survey is people who have regularly consumed video games as a piece of entertainment media. Hence, the participants of the survey have been recruited from the member base of Blueshell Esports[11], the gaming and esports association of the University of Twente.

The survey starts with an introduction page that introduces the participants to the project, its goals and includes a consent form. Then the participants are further informed on the general structure of the survey and asked to specify their familiarity with the Fallout 4 [5] lore and setting. This is asked because one of the metrics of evaluation is based on the suitability of the text within the Fallout 4 in-game world and we would like to have a better understanding of the credibility of the participants' evaluation on this metric. Participants indicate their familiarity using a numerical rating scale ranging from 1 (Not at all familiar) to 5 (Very familiar). Following this, the survey proceeds to part 1.

Part 1 contains prompt-response pairs for participants to rate on various metrics. A prompt-response pair contains a prompt and a response that has been either written by a human in response to the prompt (the original response that appears in the game) or generated by the developed language model using the prompt as an input. There are 20 of these pairs: 10 of these pairs include a human-written response and 10 of these pairs include a response generated by the language model. The prompts in these pairs are all

[11]Link to the association website: `https://esa-blueshell.nl/`

human-written. The metrics these pairs are asked to be rated on are:

- **Coherence** which indicates whether the response is grammatically correct and semantically meaningful;
- **Relevance** which represents whether the response is appropriate for the prompt;
- **Human-likeness** which is the likeliness of the response being written by a human being;
- **Fittingness** which stands for how much the response fits the Fallout 4 video game world.

The ratings are done on numerical scales ranging from 1 to 5 where 1 is labeled as the weakest occurrence of the metric, and 5 is labeled as the strongest occurrence of the metric. For instance, in the case of rating Coherence, 1 is labeled as "Not Coherent" and 5 is labeled as "Coherent". After getting informed on how part 1 is structured, the participants proceed to the rating pages. Each page contains a single prompt-response pair and participants are asked to rate it on the 4 aforementioned metrics with no indication of whether the pair is human-written or generated by the developed language model.

Part 2 contains 20 prompt-response pairs for participants to identify the affect exhibited by them. All the responses in this part have been generated by the language model with the corresponding prompt taken in as the input. For each of the pairs, participants are asked to pick the affect that is closest to being exhibited in the response from the list of 5 affects, namely, Joy, Anger, Sadness, Fear, and Disgust. According to a number of cross-cultural studies, the 5 affects used here are universally recognized the basic displayed emotions [16] [15] [32], hence we are able to assume this set of 5 affects covers all affects on a basic level. There are 2 reasons why this approach was chosen over scoring the responses on VAD scores. Firstly, from our own experience, assigning VAD values to a given response is harder than judging a single word. In more detail, while the Valence value was simpler to be deduced, scoring the Arousal and Dominance level of the responses ended up being blurry and heavily subjective. Secondly, the additional time needed for consistently judging the response on VAD space would force us to reduce the number of pairs in this part in order to maintain the temporal brevity of the survey. Using our preferred method keeps the task of this part understandable and allows the survey to be less time-consuming for the participants. After getting introduced to how part 2 is structured, the participants are taken to a page where they are asked to judge 20 prompt-response pairs with no indication of the target affect the responses were generated on.

## 6.2 Prompt-response Population

In order to populate the survey with the prompt-response samples, 40 prompts are randomly selected from the Fallout 4 dialogue dataset with the criterion of containing more

than 4 tokens. 20 of these prompts are selected for part 1, while the other 20 are used for part 2. In part 1, 10 of the select prompts are marked to be in human-written prompt-response pairs, thus they are paired up with the responses they appear with in the dataset. The rest of the prompts used in part 1 and part 2 are used as inputs for the developed language model to generate responses.

In addition to the prompt, the language model also takes a target affect in the shape of a 3-tuple containing values in VAD space as input. Since part 2 asks the participants to classify the generated responses using the set of 5 affects - Joy, Anger, Sadness, Disgust and Fear -, the language model should take a target affect value that represents any affect in this set in order to generate text that exhibits the same affect. To obtain the 3-tuples in VAD space that correspond to the said 5 affects, we have consulted the NRC: Valence, Arousal, Dominance lexicon [38]. For instance, the VAD values of the word Joy in this lexicon are 0.98 for Valence, 0.824 for Arousal, and 0.794 for Dominance. Therefore, to generate a response that expresses Joy, the language model is given these values as the target affect. Table 6 shows the VAD values of all the 5 affects found in the NRC: VAD lexicon. For part 1, each of the 5 affects has been used as an input for 2 of the response generations, whereas in part 2, each affect value has been used as an input for 4 of the response generations. As a result, in both parts there is an equal degree of representation of the 5 affects. The model-generated and human-written prompt-response pairs used in the survey are shown in Appendix A.

| Affect | Valence | Arousal | Dominance |
|---|---|---|---|
| Joy | 0.98 | 0.824 | 0.794 |
| Anger | 0.167 | 0.865 | 0.657 |
| Sadness | 0.052 | 0.775 | 0.317 |
| Fear | 0.073 | 0.840 | 0.293 |
| Disgust | 0.052 | 0.288 | 0.164 |

Table 6: 5 Affects with their VAD values from NRC: VAD lexicon.

### 6.3 Survey Analysis

The results of part 1 contain ratings on 4 metrics for 20 prompt-response pairs. The median value of each metric rating for each pair is obtained by calculating the median across all participants' scores. Since the data being used here is classified as ordinal data, medians are the recommended way of representing a set of data points [55]. As a result of this operation, each pair has 4 median ratings attached to them, in addition to the identifier that denotes if they are human-written or model-generated. For each metric, the Mann-Whitney U test [35] is performed on the median ratings of that metric to determine whether the model-generated text is comparable to the human-written one. The null hypothesis for each metric test is that the median rating of the model-generated responses is

from the same distribution as the median rating of the human-written responses on that metric. This implies that the test is a two-tailed test and the chosen significance level alpha = 0.05. The Mann-Whitney U test, also known as the Wilcoxon rank-sum test, is one of the recommended choices to test significance when it comes to ordinal data [55]. By running the test on the Coherence, Relevance, and Fittingness metric, we answer the research questions **RQ2a**, **RQ2b**, and **RQ2c**, respectively. Performing the same test on the Human-likeness metric provides a general answer to the research question **RQ2**. For the Fittingness metric, the test is ran for only the results from participants who assigned 3 or higher on the familiarity scale. In addition to the tests, the median, mode and the frequency distribution of the ratings obtained from the part 1 of the survey is also taken a look at for analyzing how responses generated with different affects perform in the evaluated metrics.

The results of part 2 contain the classification of 20 prompt-response pairs into classes of 5 affects. All of these values are put on a confusion matrix where the columns represent assigned affects and the rows stand for the affects the responses were generated on. Using this matrix, the overall accuracy of the affective correctness can be determined together with the recall and precision of individual affects. These values help us to answer the research question **RQ3**.

### 6.4 Survey Distribution

The participants of the survey has been recruited through the Discord server of the Blueshell Esports student association. A message was sent that shared the general scope of the research project and what is roughly expected of the participants. Based on this information, the association members could choose to participate in the survey or not. The contact with the members who confirmed their interest in filling the survey was made through direct messaging on Discord platform. Using this communication channel, the survey was made public on 17.12.2021 and lasted a full week (till 24.12.2021). A total of 26 individuals have filled out the survey and 15 of those participants indicated their familiarity (with Fallout 4 setting) score to be 3 out of 5 or higher.

# 7 Results

This section will display the results of the survey and analysis over those, coupled with takeaways we expounded upon in the following Discussions section.

## 7.1 Survey Part 1 Results

A Mann-Whitney U test was performed on the median ratings of each metric for all prompt-response pairs. Since the sample sizes for both human-written and model-generated responses are 10 and the significance level $\alpha = 0.05$, the critical $U_{crit}$ value for this two-tailed test is 23 [37]. Table 7 displays the necessary values regarding the tests for each of the metrics. In this table, the results of the Fittingness metric test are also shown for the participants with familiarity scores higher than or equal to 3. As it can be seen from the results, since the $U$ value for all these metrics is below $U_{crit}$, the null hypotheses for all the metrics are rejected. This indicates that the rating of the model-generated responses is significantly different from the rating of the human-written responses for the metrics Coherence, Relevance, Human-likeness, and Fittingness. Looking at the $U_{model}$ and $U_{human}$ values, it is deduced that the overall rated performance of the language model is inferior to the human-writers in all the mentioned metrics. This inferiority coupled with the significant difference that the Mann-Whitney U tests' results indicated, implies that the model-generated responses perform significantly worse than the human-written ones across all the metrics that they were evaluated on. When compared, the performance of the model-generated texts is the best in Fittingness and worst in Relevance. The results of the former is higher when the test was performed only on the ratings done by participants who have indicated their familiarity level to be 3 or higher.

| **Metric** | $U_{model}$ | $U_{human}$ | $U$ |
|---|---|---|---|
| Coherence | 8.5 | 91.5 | 8.5 |
| Relevance | 2 | 98 | 2 |
| Human-likeness | 10 | 90 | 10 |
| Fittingness | 12.5 | 87.5 | 12.5 |
| Fittingness (familiar) | 18 | 82 | 18 |

TABLE 7: Mann-Whitney U test reportables for all metrics.

In addition to the generalizing Mann-Whitney U test results, Tables 8, 9, 10, and 11 provides the target affects that the responses were generated on together with descriptive statistics of the ratings for each model-generated prompt-response pair on Coherence, Relevance, Human-likeness and Fittingness metrics, respectively. The descriptive statistics include the median, mode and the frequency distributions of the ratings. The human-written counterparts of these tables are given in Appendix B. It is important to note that the IDs displayed on the these tables (both model-generated and human-written) match the IDs of the pairs shown in Appendix A. From the results displayed in Table 8, it can be

seen that the prompt-response pairs with the target affect of Joy have the highest median and mode in Coherence ratings, as opposed to the pairs with the target affect of Anger which has the lowest ratings among all the affects. The pairs with the target affect Sadness are the second-best performing in the same metric, whereas the pairs with Disgust or Fear as their target affect have inconsistent median and mode values ranging from 2 to 5. Like the Coherence metric, the pairs with the target affect Joy also perform the best in the Relevance metric, albeit to a lower extent. Most of the pairs have the median and mode values that are equal to two or three with the exception of the pairs with the IDs 8 and 6 which perform above average with median and mode values equal to 4.

| ID | Target Affect | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 |
| **5** | Anger | 2 | 2 | 6 | 11 | 5 | 3 | 1 |
| **7** | | 3 | 4 | 1 | 8 | 6 | 11 | 0 |
| **2** | Disgust | 2 | 2 | 4 | 11 | 7 | 2 | 2 |
| **10** | | 4 | 5 | 0 | 2 | 7 | 7 | 10 |
| **3** | Fear | 2 | 2 | 6 | 11 | 6 | 3 | 0 |
| **8** | | 4.5 | 5 | 0 | 1 | 2 | 10 | 13 |
| **4** | Joy | 4 | 4 | 0 | 0 | 3 | 12 | 11 |
| **6** | | 5 | 5 | 0 | 1 | 0 | 10 | 15 |
| **1** | Sadness | 4 | 4 | 0 | 3 | 3 | 15 | 5 |
| **9** | | 4 | 5 | 0 | 1 | 3 | 10 | 12 |

TABLE 8: Target Affect, Median, Mode and the Frequency Distribution of the ratings of model-generated prompt-response pairs for the Coherence metric.

| ID | Target Affect | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 |
| **5** | Anger | 2 | 2 | 10 | 11 | 3 | 2 | 0 |
| **7** | | 3 | 3 | 4 | 1 | 15 | 5 | 1 |
| **2** | Disgust | 3 | 3 | 4 | 7 | 8 | 6 | 1 |
| **10** | | 2 | 2 | 2 | 13 | 7 | 4 | 0 |
| **3** | Fear | 2 | 2 | 5 | 10 | 7 | 3 | 1 |
| **8** | | 4 | 4 | 2 | 1 | 3 | 15 | 5 |
| **4** | Joy | 3 | 3 | 0 | 8 | 9 | 8 | 1 |
| **6** | | 4 | 4 | 0 | 6 | 2 | 14 | 4 |
| **1** | Sadness | 2 | 2 | 9 | 11 | 5 | 1 | 0 |
| **9** | | 3 | 2 | 0 | 8 | 6 | 8 | 4 |

TABLE 9: Target Affect, Median, Mode and the Frequency Distribution of the ratings of model-generated prompt-response pairs for the Relevance metric.

The performance of the pairs in the Human-likeness metric follows almost the same trend as the Coherence metric where the prompt-response pairs generated with Sadness or Joy as their target affect have performed the best and the Anger pairs obtained the lowest ratings. This indicates slight correlation between the results of these two metrics. Finally, in the Fittingness metric, the prompt-response pairs have been rated to perform relatively similarly with the Fear pairs having slightly better results.

| ID | Target Affect | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 |
| **5** | Anger | 2 | 1 | 11 | 9 | 4 | 2 | 0 |
| **7** | | 3 | 2 | 1 | 9 | 9 | 5 | 2 |
| **2** | Disgust | 2 | 2 | 7 | 10 | 5 | 3 | 1 |
| **10** | | 3 | 3 | 1 | 7 | 13 | 4 | 1 |
| **3** | Fear | 2 | 2 | 7 | 12 | 6 | 0 | 1 |
| **8** | | 4 | 4 | 0 | 1 | 8 | 10 | 7 |
| **4** | Joy | 4 | 4 | 0 | 7 | 4 | 10 | 5 |
| **6** | | 4 | 4 | 1 | 1 | 2 | 18 | 4 |
| **1** | Sadness | 4 | 4 | 0 | 5 | 6 | 12 | 3 |
| **9** | | 4 | 4 | 0 | 3 | 7 | 13 | 3 |

TABLE 10: Target Affect, Median, Mode and the Frequency Distribution of the ratings of model-generated prompt-response pairs for the Human-likeness metric.

| ID | Target Affect | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 |
| **5** | Anger | 3 | 3 | 2 | 8 | 10 | 6 | 0 |
| **7** | | 4 | 4 | 0 | 5 | 6 | 12 | 3 |
| **2** | Disgust | 3 | 3 | 1 | 7 | 11 | 6 | 1 |
| **10** | | 3.5 | 4 | 2 | 3 | 8 | 9 | 4 |
| **3** | Fear | 4 | 4 | 1 | 4 | 7 | 12 | 2 |
| **8** | | 4 | 4 | 0 | 0 | 6 | 11 | 9 |
| **4** | Joy | 3 | 3 | 1 | 3 | 11 | 7 | 4 |
| **6** | | 4 | 4 | 0 | 2 | 8 | 10 | 6 |
| **1** | Sadness | 4 | 4 | 0 | 1 | 6 | 14 | 5 |
| **9** | | 3 | 3 | 1 | 2 | 14 | 6 | 3 |

TABLE 11: Target Affect, Median, Mode and the Frequency Distribution of the ratings of model-generated prompt-response pairs for the Fittingness metric.

## 7.2 Survey Part 2 Results

Table 12 contains the confusion matrix together with the accuracy (the fraction of correctly classified samples among all generated samples) of the affective generation, precision (the fraction of correctly classified samples among all samples generated with that affect) and recall (the fraction of correctly classified samples among all samples that have been classified as that affect) of individual affects.

| | **Disgust** | **Anger** | **Fear** | **Joy** | **Sadness** | **Precision** |
|---|---|---|---|---|---|---|
| **Disgust** | 20 | 32 | 5 | 23 | 24 | 0.19 |
| **Anger** | 22 | 34 | 13 | 2 | 33 | 0.33 |
| **Fear** | 21 | 50 | 15 | 11 | 7 | 0.14 |
| **Joy** | 4 | 3 | 10 | 79 | 8 | 0.76 |
| **Sadness** | 16 | 23 | 21 | 4 | 40 | 0.38 |
| **Recall** | 0.24 | 0.24 | 0.23 | 0.66 | 0.36 | 0.36 |

TABLE 12: Confusion matrix where the columns stand for classified samples and the rows stand for the generated samples. The cell at the bottom-right contains the accuracy of the affective generation.

From Table 12, it can be seen that the accuracy of the model-generated text exhibiting the correct affect is 36% which is quite low. When looking at the precision of the individual affects, the Joy affect performs significantly better than the rest where 76% of all samples with this affect has been correctly classified. This superiority also shows itself among the Recall values with 66% of the samples that are classified with the Joy affect being samples that are generated by the model using the same affect. Sadness affect is the follow-up to Joy in both metrics with much lower values - 36% Recall and 38% Precision. The rest of the affects perform really poorly with precision and recall values below 25% with the exception of Anger whose precision is close to Sadness's with 33%. Additionally, the samples with the affects Disgust and Fear have been predominantly misclassified as Anger affect. In general, the affect Anger contains the largest amount of misclassification (by amount, not by proportion) among all the affects. On the same token, the samples generated with the affect Fear has been misclassified the most amount of times both by amount and proportion.

# 8 Discussion

In this section, we highlight the important findings from the results and we point out the problems with our research that either explains the reasons behind the findings or the lack thereof. The mentioned discussion is all contained under the Findings subsection, only to be followed by the Recommendations subsection which includes future work that could address the stated problems with our project.

## 8.1 Findings

As is previously mentioned, the model-generated responses performed the best in the Fittingness metric. This was expected due to two main reasons. Firstly, for the text to perform badly in this metric it would need to contain words or concepts that exist outside the game world. This is not likely to happen in this project because of the fact that Fallout 4's setting resembles the real world. Secondly, the use of words that are specific to the game world boosts the perceived suitability level of the words in the Fallout 4 setting. As opposed to the Fittingness metric, we expected the generated responses to perform worst in the Human-likeness metric simply because of the fact that it relies heavily on all the other metrics. For instance, low Coherence (poor grammar or meaningless gibberish), low Relevance (irrelevant response), or low Fittingness (use of words that do not fit into the Fallout 4 world) can all be telling signs of the response not being written by a human writer of the game. However, this compound metric has the second-best comparative performance measure of the model-generated responses. When it comes to the performance of the text in the Coherence and Relevance metrics, our expectations were set high because the Affect-ON approach [11] was also evaluated on these metrics. The conclusion of the paper suggested that the generated text when compared to human-written one did not suffer in syntactic coherence and appropriateness (which corresponds to Coherence and Relevance metrics of this project, respectively). However, these expectations were not met: the model-generated text performed the worst in these metrics and compared very poorly to the human-written responses.

Pinpointing the exact reason behind the low performance of the generated text in the above metrics is unclear due to the oversights in how this research was conducted. First and foremost, the FallouGPT - GPT-2 model that has been fine-tuned using the Fallout 4 dialogue dataset - was not evaluated on its own. This makes it hard to identify which design step is the source of the lackluster performance - the Fallout 4 dialogue dataset and the capabilities of GPT-2 or the affective extension and Top-K sampling. Secondly, part 1 of the evaluation survey of the responses included only the response generated with both affective extension and sampling method. This, in turn, also doesn't help to locate the component of our language model that performed badly.

When comparing the precisions of different affects on the Table 12, we can see that the

generated text with Joy target affect gets classified under the same category more often than the rest of the affects. We think this is caused by the fact that Joy is the only affect with high Valence. In comparison, the rest of the affects have much lower Valence values, thus creating the contrast that makes it simpler to distinguish Joy from the rest. A similar contrast in the Arousal values exists in the affect Disgust, however, this does not help the identification of the affect in the same efficient manner. This indicates that the participants were able to distinguish between high and low Valence better than high and low Arousal. Moreover, Anger is the affect that most responses have been wrongly classified into. We suspect that this is the result of the survey participants using this class as the umbrella affect for all negative affects. In other words, when identifying low Valence, the participants were more drawn to classify the response at hand into the Anger affect than any other. This claim is further supported by: the fact that Disgust, Fear, and Sadness affects (all having low Valence value) were misclassified as Anger the most; and the fact that affect Joy was misclassified as Anger the least. This claim if true, would also imply that the participants had an easier time identifying the Valence value of the target affect to be high or low compared to the other dimensions of the VAD space.

For affective correctness, we believe that the method in which we chose to measure it has its flaws. First and foremost, the choice to ask the participants to classify the prompt-response pairs into 5 basic affects made for a very limited representation of how the language model can express affect. This is further worsened by the limited variation in the Valence, Arousal, and Dominance values the 5 affects had. For example, the affects Sad and Fear had very similar values for all 3 of the VAD dimensions. These factors reduce our confidence in whether the way we evaluated the affective correctness of the model-generated texts is valid or not.

## 8.2 Recommendations

We believe that the first step of the future works of this project lays in the better conduct of the research. As was stated above, the way in which this research was carried out does not allow us to narrow down exactly what the weak link is. The first thing we suggest is that the FallouGPT should be evaluated on the metrics Coherence, Relevance, Human-likeness, and Fittingness on its own. This can be done by inserting a round of intermediate evaluation of the responses generated by the fine-tuned GPT-2 model. This would allow the researchers to either rule out that the cause of poor performance lies at this step or locate and attempt to better the performance of the FallouGPT before extending the generation pipeline with affective extension and sampling. Having an intermediate evaluation is obviously time-consuming, thus an alternative exists that helps the same issue where responses generated just by FallouGPT are also included in part 1 of the final evaluation survey. This approach would enable the results to indicate whether the responses generated with the affective extension and Top-K sampling are better or worse than the ones generated without the two. By the same token, responses generated with and without

Top-K sampling can also be included in part 1 of the survey to further narrow down the cause of low performance. Needless to say, the inclusion of varied model-generated responses would increase the number of prompt-response pairs to be rated by participants which increases the total time spent on the survey, however, the results that this method would bring forward justify the added time and effort spent by the survey participants.

Granted that the way in which the research is conducted enables the researchers to identify the potential source of the lackluster performance in Coherence, Relevance, Humanlikeness, and Fittingness, we think that the underlying issue can be found in one or more of the following factors: Base model selection, Fine-tuning parameters, Generation parameters and affective extension.

When it comes to the base model selection, we see a definite improvement in employing a larger and better performing base language model instead of GPT-2. The options for this upgrade include GPT-2 XL [52], GPT-Neo [8], T5 [48], Turing-NLG [49] and GPT-3 [10]. As was mentioned before, the alternatives were ruled out for this project for two main reasons - the computational infeasibility of fine-tuning the model with the resources we had and the models being closed-source. Though with more computational power and sufficient access to these models, the use of these models could boost performance measurements in the stated metrics.

Fine-tuning the base language model is another point of attention that we believe to be relevant for improving the general quality of the generated text. In our case, the selection of the parameters followed the default values provided by the Hugging Face framework with some tweaks that were partially caused by the limited capabilities of the Google Colab environment. We think that performing a grid-search on parameters like the number of epochs or learning rate with subjective evaluation could allow the researchers to identify the best performing parameters for dialogue generation that fits within the setting of the chosen video game. The same claim can be made on the optimization of the generation parameters. Even though in this project we have modified the generation parameters slightly from their default or recommended (from literature) values to eliminate some of the artifacts we have faced, this procedure can be done more efficiently and systematically by establishing subjective metrics and evaluating texts generated with varying parameters on the same metrics. Applying this method would result in parameter selection that is done in a much more comprehensive manner and parameter values that are subjectively the best for the affective dialogue generation.

Finally, the affective extension which follows the implementation of the Affect-ON approach taken by Bucinca et al. [11] can be the underperforming component of the language model. If this is the case, this would mean that the evaluation done for the syntactic coherence (Coherence) and appropriateness (Appropriateness) metrics for Affect-ON is faulty

and does not reflect the truth. In more detail, the rating for these metrics has been done using human participants and it has been done on a scale ranging from 0 to 2. This 3-level scale differs from the rating scales we used in the evaluation survey which had 5 levels. According to McKelvie [36], the 5-category is deemed most reliable and rating scales with fewer than 5 categories might result in a loss of discriminative power and validity. This can explain the differences in the overall results for seemingly similar evaluations. We believe the potential remedy for a better affective extension implementation lies in trying different approaches like Affect-LM [22] or Affect Control Theory [1] to infuse the generated text with the desired affects (if the affective extension is indeed the source of the issue).

As discussed above, the method in which the affective correctness of the model-generated response was measured made for a limited representation of the affective generation. In our opinion, the most straightforward way to mend this issue is to ask the participants to rate the pairs in VAD (Valence, Arousal, Dominance) dimensions rather than classifying them into 5 basic affect categories. This form of rating would make for a more symmetric method of evaluation since the affective extension also works with VAD values itself. The choice to go for our approach was caused by the difficulty of objective VAD rating and how it would have increased the time spent on the survey by a considerable amount.

# 9 Conclusion

With the large amount of textual content in the form of character dialogues and the necessity for a high level of immersion, video games of today require more writing than ever before. A tool that can aid this process or completely automate it can reduce the pre-production time that goes into writing every single line. With this project, we attempted to build this tool by developing a language model that can generate affective dialogue for video games, in our case the popular RPG video game Fallout 4 [5]. We utilized the existing dialogue from the game to fine-tune GPT-2 base model [47] and implemented the Affective Extension which incorporated the sense of affect into text generation. This extension closely followed the implementation of AffectON [11]. Using the fine-tuned GPT-2 and the extension, we developed a generation pipeline that promised an affective dialogue generation. The mentioned steps in building the language model directly answer the research question **RQ1**.

To evaluate this language model, we conducted a survey and performed statistical analysis to obtain results. Our results indicate that the model-generated responses compare rather poorly to human-written ones in:

- **(RA2a)** - Grammatical correctness and Semantic meaningfulness (Coherence metric);
- **(RA2b)** - Response's appropriateness for the given prompt (Relevance metric);
- **(RA2c)** - Response's suitability for Fallout 4's lore and setting (Fittingness metric).

These takeaways combined with the results we obtained on the analysis of the Human-likeness metric answer the **RQ2**:

- **(RA2)** - The model-generated responses compare poorly to human-written ones.

Additionally, the accuracy of the model-generated text exhibiting the correct affect is quite low, resides at 36%. This answers the **RQ3**:

- **(RA3)** - The responses generated by the language model exhibit the given target affect inaccurately.

The reason behind the lackluster performance of the developed language model in terms of the general quality of the generated dialogue is not possible to accurately identify due to how the research was conducted. Evaluation of the generated response with and without affective extension would help to locate the underlying source of the issue. Keeping this uncertainty in mind, we suggest several recommendations to improve the performance of the language model. Firstly, we think that a pre-trained language model with more parameters and higher levels of training such as GPT-3 [10], [48], Turing-NLG [49] etc. would be a good upgrade. Secondly, the method in which the fine-tuning and generation parameters were chosen can be made more systematic and comprehensive with the usage

of the grid-search method. Thirdly, if the source of low performance is identified to be the affective extension, alternative implementations like Affect-LM [22] or Affect Control Theory [1] can be employed to potentially improve the results. With regards to the affective correctness, we find the method in which we have evaluated the accuracy of exhibiting the target affect flawed since the basic 5 affects used for the classification are a limited representation of affect that can be sensed from the text. To mend this, we suggest asking the survey participants to rate the responses in Valence, Arousal, and Dominance dimensions rather than classify them in the mentioned 5 categories in the evaluation survey.

# References


[1] Nabiha Asghar et al. *Generating Emotionally Aligned Responses in Dialogues using Affect Control Theory*. 2020. arXiv: 2003.03645 [cs.CL].

[2] Atari. *Neverwinter Nights (PC Version)*. Video game. BioWare, Obsidian Entertainment, 2002.

[3] Y. Bengio, P. Simard, and P. Frasconi. "Learning long-term dependencies with gradient descent is difficult". In: *IEEE Transactions on Neural Networks* 5.2 (1994), pp. 157–166. DOI: 10.1109/72.279181.

[4] Thérèse Bergsma, Judith van Stegeren, and Mariët Theune. "Creating a Sentiment Lexicon with Game-Specific Words for Analyzing NPC Dialogue in The Elder Scrolls V: Skyrim". English. In: *Workshop on Games and Natural Language Processing*. Marseille, France: European Language Resources Association, May 2020, pp. 1–9. ISBN: 979-10-95546-40-5. URL: https://aclanthology.org/2020.gamnlp-1.1.

[5] Bethesda Softworks. *Fallout 4 (PC Version)*. Video game. Bethesda Game Studios, 2015.

[6] Bethesda Softworks. *Fallout: New Vegas (PC Version)*. Video game. Obsidian Entertainment, 2010.

[7] Bethesda Softworks. *The Elder Scrolls V: Skyrim (PC Version)*. Video game. Bethesda Game Studios, 2011.

[8] Sid Black et al. *GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow*. Version 1.0. Mar. 2021. DOI: 10.5281/zenodo.5297715. URL: https://doi.org/10.5281/zenodo.5297715.

[9] Margaret M. Bradley and Peter J. Lang. "Affective Norms for English Words (ANEW): Instruction Manual and Affective Ratings". In: *Technical Report, C-1*. The Center for Research in Psychophysiology, University of Florida, 1999.

[10] Tom Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[11] Zana Bucinca et al. "AffectON:Incorporating Affect Into Dialog Generation". In: *IEEE Transactions on Affective Computing* (2020), pp. 1–1. DOI: 10.1109/TAFFC.2020.3043067.

[12] Paweł Budzianowski and Ivan Vulić. "Hello, It's GPT-2 - How Can I Help You? Towards the Use of Pretrained Language Models for Task-Oriented Dialogue Systems". In: *Proceedings of the 3rd Workshop on Neural Generation and Translation*. Hong Kong: Association for Computational Linguistics, Nov. 2019, pp. 15–22. DOI: 10.18653/v1/D19-5602. URL: https://aclanthology.org/D19-5602.

[13] Orianna Demasi, Yu Li, and Zhou Yu. "A Multi-Persona Chatbot for Hotline Counselor Training". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 3623–3636. DOI: 10.18653/v1/2020.findings-emnlp.324. URL: https://aclanthology.org/2020.findings-emnlp.324.

[14] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

[15] Paul Ekman and Wallace V Friesen. "Constants across cultures in the face and emotion." In: *Journal of personality and social psychology* 17.2 (1971), p. 124.

[16] Paul Ekman, E Richard Sorenson, and Wallace V Friesen. "Pan-cultural elements in facial displays of emotion". In: *Science* 164.3875 (1969), pp. 86–88.

[17] Electronic Arts. *Dragon Age: Origins (PC Version)*. Video game. BioWare, 2009.

[18] Electronic Arts. *Mass Effect 3 (PC Version)*. Video game. BioWare, 2012.

[19] Angela Fan, Mike Lewis, and Yann Dauphin. "Hierarchical Neural Story Generation". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 889–898. DOI: 10.18653/v1/P18-1082. URL: https://aclanthology.org/P18-1082.

[20] Bjarke Felbo et al. "Using millions of emoji occurrences to learn any-domain representations Fgptfor detecting sentiment, emotion and sarcasm". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 1615–1625. DOI: 10.18653/v1/D17-1169. URL: https://aclanthology.org/D17-1169.

[21] Jamie Fraser, Ioannis Papaioannou, and Oliver Lemon. "Spoken Conversational AI in Video Games: Emotional Dialogue Management Increases User Engagement". In: *Proceedings of the 18th International Conference on Intelligent Virtual Agents*. IVA '18. Sydney, NSW, Australia: Association for Computing Machinery, 2018, pp. 179–184. ISBN: 9781450360135. DOI: 10.1145/3267851.3267896. URL: https://doi.org/10.1145/3267851.3267896.

[22] Sayan Ghosh et al. "Affect-LM: A Neural Language Model for Customizable Affective Text Generation". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, July 2017, pp. 634–642. DOI: 10.18653/v1/P17-1059. URL: https://aclanthology.org/P17-1059.

[23] Tushar Goswamy et al. “Adapting a Language Model for Controlled Affective Text Generation”. In: *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 2787–2801. DOI: 10.18653/v1/2020.coling-main.251. URL: https://aclanthology.org/2020.coling-main.251.

[24] Donghoon Ham et al. “End-to-End Neural Pipeline for Goal-Oriented Dialogue Systems using GPT-2”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 583–592. DOI: 10.18653/v1/2020.acl-main.54. URL: https://aclanthology.org/2020.acl-main.54.

[25] David Helbig, Enrica Troiano, and Roman Klinger. “Challenges in Emotion Style Transfer: An Exploration with a Lexical Substitution Pipeline”. In: *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*. Online: Association for Computational Linguistics, July 2020, pp. 41–50. DOI: 10.18653/v1/2020.socialnlp-1.6. URL: https://aclanthology.org/2020.socialnlp-1.6.

[26] Kyle Hilliard. *Bethesda completes recording of Fallout 4's 111,000 lines of dialogue*. 2015. URL: https://www.gameinformer.com/b/news/archive/2015/09/05/bethesda-completes-recording-of-fallout-4-39-s-111-000-lines-of-dialogue.aspx (visited on 12/09/2021).

[27] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[28] Zhiheng Huang, Wei Xu, and Kai Yu. “Bidirectional LSTM-CRF Models for Sequence Tagging”. In: *Proceedings of the 21st International Conference on Asian Language Processing* (2015).

[29] Christopher Kerr and Duane Szafron. “Supporting Dialogue Generation for Story-Based Games”. In: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 5.1 (Oct. 2009), pp. 154–160. URL: https://ojs.aaai.org/index.php/AIIDE/article/view/12371.

[30] Lee Kezar. “Mixed Feelings: Natural Text Generation with Variable, Coexistent Affective Categories”. In: *Proceedings of ACL 2018, Student Research Workshop*. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 141–145. DOI: 10.18653/v1/P18-3020. URL: https://aclanthology.org/P18-3020.

[31] Wei-Jen Ko and Junyi Jessy Li. “Assessing Discourse Relations in Language Generation from GPT-2”. In: *Proceedings of the 13th International Conference on Natural Language Generation*. Dublin, Ireland: Association for Computational Linguistics, Dec. 2020, pp. 52–59. URL: https://aclanthology.org/2020.inlg-1.8.

[32] Robert W Levenson. “Autonomic specificity and emotion”. In: *Handbook of affective sciences* 2 (2003), pp. 212–224.

[33] LucasArts, Electronic Arts. *Star Wars: Knights of the Old Republic (PC Version)*. Video game. BioWare, Obsidian Entertainment, Aspyr Media, 2003.

[34] Nikolay Malkin et al. "GPT Perdetry Test: Generating new meanings for new words". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, June 2021, pp. 5542–5553. DOI: 10.18653/v1/2021.naacl-main.439. URL: https://aclanthology.org/2021.naacl-main.439.

[35] Henry B Mann and Donald R Whitney. "On a test of whether one of two random variables is stochastically larger than the other". In: *The annals of mathematical statistics* (1947), pp. 50–60.

[36] Stuart J McKelvie. "Graphic rating scales—How many categories?" In: *British Journal of Psychology* 69.2 (1978), pp. 185–202.

[37] Roy C Milton. "An extended table of critical values for the Mann-Whitney (Wilcoxon) two-sample statistic". In: *Journal of the American Statistical Association* 59.307 (1964), pp. 925–934.

[38] Saif Mohammad. "Obtaining Reliable Human Ratings of Valence, Arousal, and Dominance for 20,000 English Words". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 174–184. DOI: 10.18653/v1/P18-1017. URL: https://aclanthology.org/P18-1017.

[39] Victoria Mozolevskaya. *Stages of game development: 6 main steps*. July 2021. URL: https://kevurugames.com/blog/6-key-stages-of-game-development-from-concept-to-standing-ovation/.

[40] Namco, Midway. *Pac-Man (Arcade Version)*. Video game. Namco, 1980.

[41] Ocelot Society. *Event[0] (PC Version)*. Video game. Ocelot Society, 2016.

[42] Nadav Oved and Ran Levy. "PASS: Perturb-and-Select Summarizer for Product Reviews". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 351–365. DOI: 10.18653/v1/2021.acl-long.30. URL: https://aclanthology.org/2021.acl-long.30.

[43] Devin Pickell. *The 7 Stages of Game Development*. Oct. 2019. URL: https://www.g2.com/articles/stages-of-game-development.

[44] Devin Pickell. *The 7 Stages of Game Development*. Oct. 2019. URL: https://www.g2.com/articles/stages-of-game-development.

[45] Procedural Arts. *Façade (PC Version)*. Video game. Procedural Arts, 2005.

[46] Alec Radford et al. “Improving language understanding by generative pre-training”. In: (2018). URL: https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf.

[47] Alec Radford et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8 (2019), p. 9. URL: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[48] Adam Roberts et al. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. Tech. rep. Google, 2019.

[49] Corby Rosset. *Turing-NLG: A 17-billion-parameter language model by Microsoft*. Feb. 2020. URL: https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/.

[50] James A Russell. “Core affect and the psychological construction of emotion.” In: *Psychological review* 110.1 (2003), p. 145.

[51] Sashank Santhanam and Samira Shaikh. “Emotional Neural Language Generation Grounded in Situational Contexts”. In: *Proceedings of the 4th Workshop on Computational Creativity in Language Generation*. Tokyo, Japan: Association for Computational Linguistics, 29 10–3 11 2019, pp. 22–27. URL: https://aclanthology.org/2019.ccnlg-1.3.

[52] Irene Solaiman et al. “Release strategies and the social impacts of language models”. In: *arXiv preprint arXiv:1908.09203* (2019).

[53] Sony Computer Entertainment. *God of War (Playstation 4 Version)*. Video game. Santa Monica Studio, 2018.

[54] Sony Computer Entertainment. *The Last of Us (Playstation 4 Version)*. Video game. Naughty Dog, 2014.

[55] Elisabeth Svensson et al. “Guidelines to statistical evaluation of data from rating scales and questionnaires”. In: *Journal of Rehabilitation Medicine* 33.1 (2001), pp. 47–48.

[56] Taito, Midway, Leisure  Allied Industries. *Space Invaders (Arcade Version)*. Video game. Taito, 1978.

[57] Valve. *Portal 2 (PC Version)*. Video game. Valve, 2011.

[58] Amy Beth Warriner, Victor Kuperman, and Marc Brysbaert. “Norms of valence, arousal, and dominance for 13,915 English lemmas”. In: *Behavior research methods* 45.4 (2013), pp. 1191–1207.

[59] Emma Welsh. *How to Write a Good Video Game Story*. Nov. 2017. URL: https://www.emwelsh.com/blog/write-good-video-game-story.

[60] Wallace Witkowski. *Videogames are a bigger industry than movies and North American sports combined, thanks to the pandemic*. 2020. URL: https://www.marketwatch.com/story/videogames-are-a-bigger-industry-than-sports-and-movies-combined-thanks-to-the-pandemic-11608654990 (visited on 12/09/2021).

[61] ZA/UM. *Disco Elysium (PC Version)*. Video game. ZA/UM, 2019.

[62] WeiSheng Zhang et al. "Multi-Turn Dialogue Generation in E-Commerce Platform with the Context of Historical Dialogue". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 1981–1990. DOI: 10.18653/v1/2020.findings-emnlp.179. URL: https://aclanthology.org/2020.findings-emnlp.179.

[63] Yizhe Zhang et al. "DIALOGPT : Large-Scale Generative Pre-training for Conversational Response Generation". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Online: Association for Computational Linguistics, July 2020, pp. 270–278. DOI: 10.18653/v1/2020.acl-demos.30. URL: https://aclanthology.org/2020.acl-demos.30.

[64] Pengfei Zhu et al. "Lingke: a Fine-grained Multi-turn Chatbot for Customer Service". In: *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*. Santa Fe, New Mexico: Association for Computational Linguistics, Aug. 2018, pp. 108–112. URL: https://aclanthology.org/C18-2024.

# Appendix A

| | Prompt | Response |
|---|---|---|
| 1 | I secured the DIA cache for you. | Enabling machine/human interface. Well done, agent. Runners will be sent to acquire contents. Reward dispensed. |
| 2 | Why do you say that? | I really shouldn't get into it. |
| 3 | Well, I've had just about all I can stomach. You've left me no choice. I'm done with you. | Is there anything I can say that will change your mind? |
| 4 | Didn't run into too much trouble coming out here, I hope? | It was terrible. Everything here is trying to kill me. |
| 5 | Have you collected any blood samples, Knight? | Yes, here you go. |
| 6 | Well, I sure do. Those green bastards are freaks of nature. | These monstrosities are just another example of man blindly taking a step forward only to wind up stumbling two steps back. |
| 7 | I have some questions first. | Of course you do. And I'm here to answer them. Shoot! |
| 8 | I feel honored to know you and to fight at your side. | You've been impressive so far. I hope that continues. |
| 9 | Oh, it's you. You're not authorized to be here. | Actually, I was looking for you. |
| 10 | Well, it's only a theory so far. | You see, the ionizing radiation left behind by the Great War kills because it damages most living things on a cellular level. |

TABLE 13: Human-written responses used in the part 1 of the survey. The leftmost column stands for the IDs of the pairs.

| | Prompt | Response |
|---|---|---|
| 1 | Yeah, that's...my...favorite subject. | {Sadness} You've been through Hell, and back again... and it's all pretty grimy. I'm just glad it's over with. |
| 2 | Ha, I knew I could count on you! | {Disgust} That's right folks... thanks. You guys made our work more fun and fun... I felt better than any bossing kid out in a shed for a drink! I can barely hold it. Thanks! For letting my hair go up. |
| 3 | Don't worry. I promise I won't get in your way. | {Fear} Oh yeah... that would help if it was all done in two things! First you help the boy, you owe the mayor your gun! |
| 4 | You just love her. Family's precious. The last thing you want to do is drive them away, because you might lose them forever. | {Joy} You know what? If I ever find out that she's addicted, I won't get involved. |
| 5 | Sir, was that really necessary? | {Anger} Yes sir! It's great advice! Thank god you didn't crash into my office while I was cleaning up. I wish there'd never had that mess left. I wish you'd reconsider, knowing that this was your fault, sir, not your god damned business, brother. |
| 6 | Can't you please take me to them? Or at least to my old house? | {Joy} No! You must be kidding. They are my family now! |
| 7 | That's an acceptable answer... and the most common. | {Anger} You don't know the answer to that question... and you don't owe me a damn damned damn thing. |
| 8 | No! You can't! He's going to kill me! | {Fear} I... I need to go! I want answers! |
| 9 | I came here for the music. | {Sadness} Oh. Well, it was nice to sit quietly and wait. |
| 10 | It's more than just a statement. It's a plea. | {Disgust} I'm afraid it doesn't need to end here. |

TABLE 14: Model-generated responses used in the part 1 of the survey. The left-most column stands for the IDs of the pairs. The affects displayed in curly brackets under the Response column stand for the target affects the responses were generated on.

| | Prompt | Response |
|---|---|---|
| 1 | There was no other choice. The Brotherhood must come first. | {Anger} You've got no problem with that. |
| 2 | You're in luck then. Covenant is the friendliest place in the whole Commonwealth. You'll see! | {Joy} Thanks to you. |
| 3 | I hate being the lowest rung on the totem pole. | {Anger} I'm sorry, Piper. I hate being the boss. |
| 4 | Well, it's definitely unique. Sort of hard to miss. | {Sadness} It's a little hard to tell if they're just sitting there, waiting for the signal to pass... or if they're actually moving in. |
| 5 | If Virgil found a way to survive there, you'll have to do the same, if you're going to follow him. | {Fear} If you find a means to use that reactor... |
| 6 | So you know 'em then. Any idea why they haven't been around much lately? | {Fear} You know what they're going to go through, I bet. They're going to need to go through me. |
| 7 | We actually did it! We took down the Prydwen! | {Joy} You did? I didn't think you'd survive that fight! |
| 8 | It needn't be swallowed at all if you ask me. Not all answers lead to bloodshed, you know. Especially that of innocent people. | {Sadness} It was the wrong thing to go through with it. |
| 9 | The boy? I wouldn't have thought him capable of such thing. | {Disgust} I don't doubt that he was suffering from... psychosis. |
| 10 | Don't just go waving it around willy-nilly. | {Fear} You don't know the difference. |

TABLE 15: First half of the model-generated responses used in the part 2 of the survey. The left-most column stands for the IDs of the pairs. The affects displayed in curly brackets under the Response column stand for the target affects the responses were generated on.

| | Prompt | Response |
|---|---|---|
| 11 | So, any other tricks up your sleeve besides making the Institute disappear? | {Disgust} You know how I feel about that. |
| 12 | I'm not sure, let me think about it. | {Joy} Well, if you do find out about it, I'd be glad to get my hands hot on it. |
| 13 | Don't change the subject. I'm the one asking questions here. I want to know about the Institute, and I want to know now. | {Sadness} I'm sorry, Piper. I don't need to apologize any more. |
| 14 | So, where do you put in the batteries on this thing? | {Fear} You're joking with me, don't you? |
| 15 | I loved my son... my real son. You're not him. It's not fair to ask me that. | {Anger} I know I'm being rude, but you don't have to kill me to protect your secret... |
| 16 | I guess he could get into real trouble if they decide he's guilty anyway. | {Anger} You know what? I'm going to kill him. I'm going to beat him to death. And I'm going to throw his body in the garbage. |
| 17 | So what do you say? Ready to go out and dissect one? | {Disgust} Sure! Here. For science experiments! You can use these for anythin', right now? I'll spare ya, I swear... but if the tin pan's dead set on destroying everything... Well then I suppose we've just had a misunderstanding... Well guess where it's at? |
| 18 | Rhys still giving you the cold shoulder? | {Disgust} No problem... we've all gotten to be human too late! But now I see it... It feels different to know he killed our mutual brother Nelson and took you by storm, all on my watch? He never would give us that type a trial until I told them it wasn. It must end! He was an evil tyrant!"> |
| 19 | Don't know what happened to the previous owners, but they're not exactly around to charge rent. An empty vault. Perfect hideout. | {Sadness} I've been to the wrong end of the alley. I saw you leaving the stash, but I don't blame you for trying to go after them. |
| 20 | You'd be putting yourself in danger. | {Joy} That sounds fair. |

TABLE 16: Second half of the model-generated responses used in the part 2 of the survey. The left-most column stands for the IDs of the pairs. The affects displayed in curly brackets under the Response column stand for the target affects the responses were generated on.

# Appendix B

| ID | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 | 5 |
| **1** | 4 | 4 | 0 | 2 | 4 | 12 | 8 |
| **2** | 5 | 5 | 0 | 0 | 0 | 3 | 23 |
| **3** | 5 | 5 | 0 | 0 | 0 | 6 | 20 |
| **4** | 5 | 5 | 0 | 0 | 0 | 5 | 21 |
| **5** | 5 | 5 | 0 | 0 | 1 | 3 | 22 |
| **6** | 5 | 5 | 0 | 1 | 2 | 7 | 16 |
| **7** | 5 | 5 | 0 | 0 | 2 | 4 | 20 |
| **8** | 5 | 5 | 0 | 0 | 1 | 7 | 18 |
| **9** | 5 | 5 | 0 | 0 | 1 | 5 | 20 |
| **10** | 5 | 5 | 0 | 0 | 1 | 9 | 16 |

TABLE 17: Median, Mode and the Frequency Distribution of the ratings of human-written prompt-response pairs for the Coherence metric.

| ID | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 | 5 |
| **1** | 5 | 5 | 0 | 0 | 2 | 8 | 16 |
| **2** | 5 | 5 | 0 | 1 | 0 | 6 | 19 |
| **3** | 5 | 5 | 0 | 2 | 0 | 5 | 19 |
| **4** | 5 | 5 | 0 | 0 | 0 | 4 | 22 |
| **5** | 5 | 5 | 0 | 0 | 2 | 3 | 21 |
| **6** | 5 | 5 | 0 | 0 | 1 | 11 | 14 |
| **7** | 5 | 5 | 0 | 0 | 0 | 4 | 22 |
| **8** | 4.5 | 5 | 0 | 0 | 3 | 10 | 13 |
| **9** | 4 | 4 | 0 | 1 | 4 | 13 | 8 |
| **10** | 4 | 4 | 0 | 3 | 6 | 10 | 7 |

TABLE 18: Median, Mode and the Frequency Distribution of the ratings of human-written prompt-response pairs for the Relevance metric.

| ID | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 | 5 |
| **1** | 3.5 | 5 | 3 | 8 | 2 | 4 | 9 |
| **2** | 5 | 5 | 1 | 0 | 2 | 4 | 19 |
| **3** | 5 | 5 | 0 | 2 | 2 | 7 | 15 |
| **4** | 5 | 5 | 0 | 0 | 0 | 8 | 18 |
| **5** | 5 | 5 | 1 | 0 | 3 | 3 | 19 |
| **6** | 5 | 5 | 0 | 0 | 2 | 8 | 16 |
| **7** | 5 | 5 | 0 | 0 | 4 | 7 | 15 |
| **8** | 4 | 5 | 0 | 2 | 2 | 10 | 12 |
| **9** | 4.5 | 5 | 0 | 2 | 3 | 8 | 13 |
| **10** | 4 | 4 | 0 | 0 | 3 | 13 | 10 |

TABLE 19: Median, Mode and the Frequency Distribution of the ratings of human-written prompt-response pairs for the Human-likeness metric.

| ID | Median | Mode | Rating Count | | | | |
|---|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 | 5 |
| **1** | 4 | 4 | 0 | 0 | 4 | 12 | 10 |
| **2** | 4 | 5 | 0 | 2 | 5 | 7 | 12 |
| **3** | 4 | 4 | 0 | 0 | 6 | 11 | 9 |
| **4** | 5 | 5 | 0 | 0 | 0 | 3 | 23 |
| **5** | 5 | 5 | 0 | 2 | 4 | 5 | 15 |
| **6** | 5 | 5 | 0 | 1 | 2 | 8 | 15 |
| **7** | 4.5 | 5 | 0 | 1 | 8 | 4 | 13 |
| **8** | 4 | 5 | 0 | 0 | 6 | 9 | 11 |
| **9** | 4 | 4 | 0 | 1 | 6 | 12 | 7 |
| **10** | 5 | 5 | 0 | 0 | 0 | 3 | 23 |

TABLE 20: Median, Mode and the Frequency Distribution of the ratings of human-written prompt-response pairs for the Fittingness metric.