Group Project:

Learning Diary Generator

Anssi Moisio & Ottilia Nikula

In this project our aim is to make a text generator that creates learning diary entries as its output. We shall first discuss the definition of creativity, the goal of the project in regards to the desired output, and which implications our definition of creativity had on our evaluation of the system. Finally, we shall discuss future work and possible improvements on the system.

We used the FACE descriptive model to assess the creativity of our program. According to this model, there are eight types of individual creative acts, which are divided into four types at the process level and the same four types at the ground level. The process level includes creative acts that produce new ways to generate and evaluate artefacts, whereas the ground level includes the acts that produce new artefacts. The four types are:

- concept, an executable program which produces an output given an input
- expression, an input-output pair generated by the program
- aesthetic measure, a function that takes an expression and outputs a value that measures the expression
- framing, a piece of text that refers to the expression, giving some metainformation about it

Our aim is that our program is creative according to one of these eight types, which is that it produces expressions creatively. The whole creative act includes also creativity from outside the program, at least by us and the researchers of OpenAI. For creative acts performed by a third party, (Colton et al., 2012) uses a bar notation. Following this notation, the collective creative act performed by the system can be written as the tuple $\langle \overline{A^g}, \overline{C^g}, E^g \rangle$, which means that there are three creative acts: the aesthetic functions created by us, the concept mostly created by (Radford et al., 2019), and the

expression created by the program.

We argue that the program is creative when it produces expressions that perform well on our aesthetic measurements, which would mean that they get similar values from them as our own example learning diary entry does. These aesthetics try to quantitatively measure how much the output looks like a learning diary and how much it addresses the lecture material given as the input.

We based our solution on a language model created by OpenAI called GPT-2 (Radford et al., 2019), or Generative Pre-trained Transformer. Transformer is an encoder-decoder model that uses multi-head attention blocks instead of recurrent neural networks. It is able to learn long-term dependencies in texts, such as named entities mentioned a few sentences apart, and use them appropriately when generating text. GPT-2 is trained on large corpus of about 40GB of text scraped from links on the website Reddit. It has not been trained specifically on any type of document or content, so it produces text that is similar to any given input text.

GPT-2 takes text as input and then generates text based on this input. In our system, we use an abstract of a journal article, an example learning diary entry from the lecture in which the article was discussed, and the abstract of the article for the next lecture as the input. The expected output is a learning diary entry that discusses the second abstract that was inputted.

After generating outputs from the language model, the program chooses the best of these according to our aesthetic functions. As the first aesthetic, we measure the similarity between the input and output texts. This is done by using Google's universal sentence encoder (Cer et al., 2018), which creates a 512-dimensional vector embedding for any short piece of text. As its name suggests, it is primarily meant for embedding sentences, but it also works for short documents. However, the quality of the embedding decreases when the length of the text increases, as the dimensionality stays constant. Creating this kind of embedding is in principle comparable to clustering the tokens of the text and then calculating the cluster centroid, because in both cases the text is embedded into a single vector, which usually has a few hundred dimensions. After creating the embeddings for each of the generated outputs as well as for the input text, we calculate the inner product of the texts and the input text. The product measures the similarity of the input and output texts, and is the value of this aesthetic function.

For the second aesthetic function, we calculate the similarity of the generated

texts to a different model text which contains phrases that indicate the writer's own thinking, such as "In my opinion…". With this function we attempt to evaluate how much the generated text resembles a learning diary as opposed to, e.g., a scientific article. After calculating the aesthetic function values, the program selects the best output text from a number of generated outputs by simply summing the aesthetic values and taking the output with the maximum of the sums.

However, we found that we are not always in agreement with the ratings given by the functions, as in some cases an entry receives a higher score, even though it resembles more of an abstract than a learning diary. The goal of these two aesthetic functions was to automatically and quantitatively evaluate questions such as "Is the writer reflecting on the lecture material", "Does this look like a learning diary?" and "Does the text indicate showcasing the writer's own thinking?". As one might have expected, this turned out to be an ambitious goal. We cannot say that these two aesthetic functions achieved the goal but we think that functions of this kind could in principle do so. When calculating the similarity values of a learning diary entry written by one of us with the aesthetic functions we get similar values (roughly 0.5) as with the computer-generated outputs whose variance on these metrics is large. Refer to the example input-output pair for the aesthetic values.

In future work, our aim would be to measure whether the text includes cross-referencing to other lectures by measuring phrases such as "in the previous lecture", "previously" etc. Furthermore, we could measure whether the output has indications of challenging or comparing theories, by searching for phrases with an x *than* y structure, where x and y are phrases containing words such as "theory", "model", "assumption", or "idea". This kind of sentence structure would indicate the comparison of two ideas. In addition, in our human evaluation assignment the readers were not presented with the abstract or the content of the lecture that the output was supposed to discuss. Therefore, we cannot know if these texts would have passed the human evaluation test, if the reader knew what the content should have included. Therefore, future work would include a more extensive human evaluation test, in which the subject gets briefed on the desired content of the output.

In conclusion, the system generates texts which passed as human generated in the human evaluation assignment, but would not necessarily do so given the context. We

created aesthetic functions, which do not always rate the text accurately and therefore a finetuning of this implementation would be required in future work.

References

Cer, D., Yang, Y., Kong, S. Y., Hua, N., Limtiaco, N., John, R. S., ... & Sung, Y. H. (2018). Universal sentence encoder. arXiv preprint arXiv:1803.11175.

Colton, S., Charnley, J. W., & Pease, A. (2011, April). Computational Creativity Theory: The FACE and IDEA Descriptive Models. In ICCC (pp. 90-95).

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8).