

CS-C1000 – Introduction to Artificial Intelligence

Computer Assignment B

William Wilkinson

March 23, 2021

Computer Exercise B: Fake Text Generation

Artificial intelligence (AI)

Alex Hern

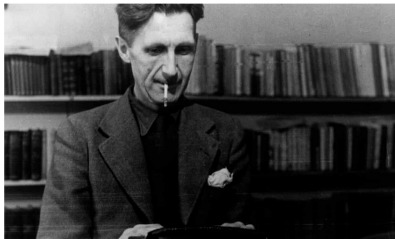
@alexhern

Thu 14 Feb 2019 17:00 GMT



New AI fake text generator may be too dangerous to release, say creators

The Elon Musk-backed nonprofit company OpenAI declines to release research publicly for fear of misuse



▲ The AI wrote a new passage of fiction set in China after being fed the opening line of *Nineteen Eighty-Four* by George Orwell (pictured). Photograph: Mondadori/Getty Images

The creators of a revolutionary AI system that can write news stories and works of fiction - dubbed "deepfakes for text" - have taken the unusual step of not releasing their research publicly, for fear of potential misuse.

OpenAI, an nonprofit research company backed by Elon Musk, Reid Hoffman, Sam Altman, and others, says its new AI model, called GPT2 is so good and the risk of malicious use so high that it is breaking from its normal practice of releasing the full research to the public in order to allow more time to discuss the capabilities of the technology and its potential for misuse.

SYSTEM PROMPT
(HUMAN-WRITTEN)

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

SYSTEM PROMPT
(HUMAN-WRITTEN)

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

MODEL COMPLETION
(MACHINE-WRITTEN,
10 TRIES)

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.


Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them – they were so close they could touch their horns.

Log into JupyterHub (a quick reminder)

Log in to:
`https://jupyter.cs.aalto.fi`



A screenshot of the JupyterHub login interface. It features an orange header bar with the text "Sign in". Below the header, there are two input fields: "Username:" and "Password:". The username field contains the text "firstname.lastname@aalo.fi". The password field is filled with dots and has a small icon of a person with a checkmark to its right. Below the password field is an orange "Sign In" button.

Using your Aalto account.

Choose the course

The screenshot shows a web browser window with the URL `jupyter.cs.aalto.fi`. The page title is "Spawner Options". It contains a list of courses, each with a radio button. The course "CS-C1000 Introduction to Artificial Intelligence" is selected, indicated by a blue dot and an orange highlight box. Below the list is a large orange button labeled "Spawn".

Course	Selected
Python: General use (JupyterLab) 0.5.9	<input type="radio"/>
Python: General use (classic notebook) 0.5.9	<input type="radio"/>
R: General use (JupyterLab) 0.5.3	<input type="radio"/>
Julia: General use (JupyterLab) 0.5.9	<input type="radio"/>
(testing) Python: General use (JupyterLab) 0.5.10	<input type="radio"/>
30E03000 Data Science for Business 2019 I	<input type="radio"/>
30E03000 Data Science for Business 2019 II	<input type="radio"/>
CS-C1000 Introduction to Artificial Intelligence	<input checked="" type="radio"/>
CS-E3100 Data Science 2018	<input type="radio"/>
CS-E3210 Machine Learning: Basic Principles 2018	<input type="radio"/>
CS-E4830 Kernel Methods in Machine Learning	<input type="radio"/>
CS-E4890 Deep Learning 2019	<input type="radio"/>
testcourse	<input type="radio"/>

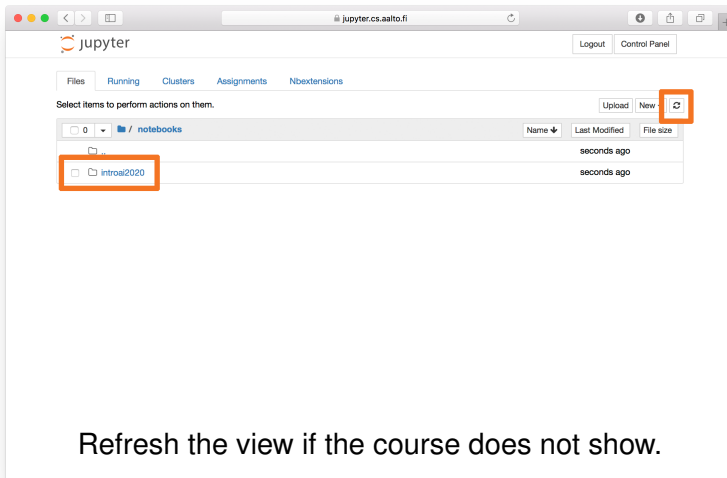
Spawn

Fetch the exercise (B this time, not A!)

The screenshot shows the JupyterLab web interface in a browser window. The address bar shows 'jupyter.cs.aalto.fi'. The interface has a top navigation bar with 'Logout' and 'Control Panel' buttons. Below this is a tabbed interface with 'Files', 'Running', 'Clusters', 'Assignments', and 'nbextensions'. The 'Assignments' tab is selected. Below the tabs, there is a dropdown menu for 'Released, downloaded, and submitted assignments for course:' with 'introai2020' selected. The main content area is divided into three sections: 'Released assignments', 'Downloaded assignments', and 'Submitted assignments'. The 'Released assignments' section contains a table with one row: 'Computer-Assignment-A' for 'introai2020'. A 'Fetch' button is located to the right of this row. The 'Downloaded assignments' and 'Submitted assignments' sections both display the message 'There are no downloaded assignments.' and 'There are no submitted assignments.' respectively. Three orange boxes and numbers highlight the steps: 1. 'Assignments' tab, 2. 'Fetch' button, and 3. 'Files' tab.

1. Choose 'Assignments', 2. Fetch, 3. Choose 'Files'

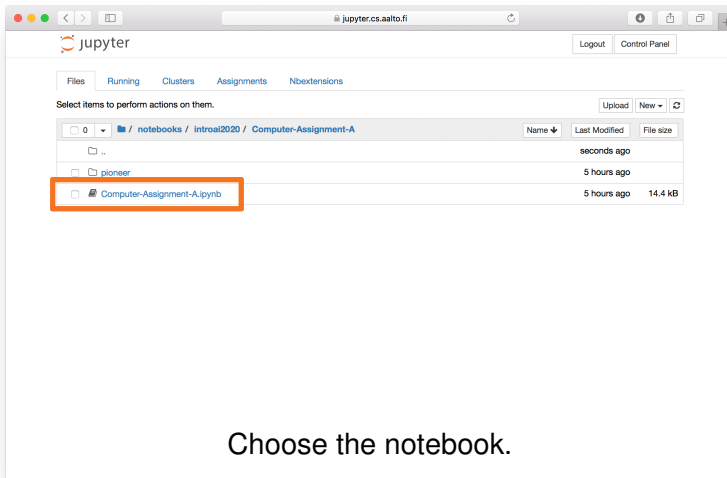
Find the notebook



The screenshot shows the JupyterLab web interface in a browser window. The address bar displays 'jupyter.cs.aalto.fi'. The interface includes a top navigation bar with 'Logout' and 'Control Panel' buttons. Below this is a tabbed interface with 'Files', 'Running', 'Clusters', 'Assignments', and 'Nbextensions'. The 'Files' tab is active, showing a file browser for the '/ notebooks' directory. A table lists files with columns for 'Name', 'Last Modified', and 'File size'. The file 'introai2020' is listed and highlighted with an orange box. Above the table, there are buttons for 'Upload', 'New', and a refresh icon, with the refresh icon also highlighted by an orange box. The text 'Select items to perform actions on them.' is visible above the table.

Refresh the view if the course does not show.

Fire it up (again, B this time)

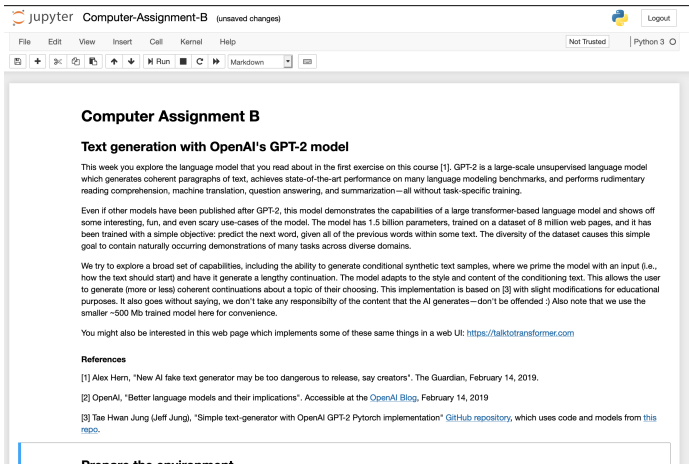


The image shows a JupyterLab web interface in a browser window. The address bar shows `jupyter.cs.aalto.fi`. The interface has a top bar with the Jupyter logo, a "Logout" button, and a "Control Panel" button. Below this is a navigation bar with tabs for "Files", "Running", "Clusters", "Assignments", and "Nbextensions". The "Files" tab is active, showing a file browser. The breadcrumb path is `/ notebooks / introai2020 / Computer-Assignment-A`. There are buttons for "Upload", "New", and a refresh icon. A table of files is displayed with columns for "Name", "Last Modified", and "File size". The file `Computer-Assignment-A.ipynb` is highlighted with an orange box.

	Name	Last Modified	File size
<input type="checkbox"/>	..	seconds ago	
<input type="checkbox"/>	pioneer	5 hours ago	
<input type="checkbox"/>	Computer-Assignment-A.ipynb	5 hours ago	14.4 kB

Choose the notebook.

Ready to run (a cell at a time)



The screenshot shows a Jupyter Notebook interface. At the top, the title bar reads 'jupyter Computer-Assignment-B (unsaved changes)' with a 'Logout' button on the right. Below the title bar is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', and 'Help'. To the right of the menu bar are status indicators: 'Not Trusted' and 'Python 3'. Below the menu bar is a toolbar with icons for saving, undo, redo, and other actions. The main content area displays a document titled 'Computer Assignment B' with the following text:

Computer Assignment B

Text generation with OpenAI's GPT-2 model

This week you explore the language model that you read about in the first exercise on this course [1]. GPT-2 is a large-scale unsupervised language model which generates coherent paragraphs of text, achieves state-of-the-art performance on many language modeling benchmarks, and performs rudimentary reading comprehension, machine translation, question answering, and summarization—all without task-specific training.

Even if other models have been published after GPT-2, this model demonstrates the capabilities of a large transformer-based language model and shows off some interesting, fun, and even scary use-cases of the model. The model has 1.5 billion parameters, trained on a dataset of 8 million web pages, and it has been trained with a simple objective: predict the next word, given all of the previous words within some text. The diversity of the dataset causes this simple goal to contain naturally occurring demonstrations of many tasks across diverse domains.

We try to explore a broad set of capabilities, including the ability to generate conditional synthetic text samples, where we prime the model with an input (i.e., how the text should start) and have it generate a lengthy continuation. The model adapts to the style and content of the conditioning text. This allows the user to generate (more or less) coherent continuations about a topic of their choosing. This implementation is based on [3] with slight modifications for educational purposes. It also goes without saying, we don't take any responsibility of the content that the AI generates—don't be offended :) Also note that we use the smaller ~500 Mb trained model here for convenience.

You might also be interested in this web page which implements some of these same things in a web UI: <https://talktotransformer.com>

References

[1] Alex Hern, "New AI fake text generator may be too dangerous to release, say creators". The Guardian, February 14, 2019.

[2] OpenAI, "Better language models and their implications". Accessible at the [OpenAI Blog](#), February 14, 2019

[3] Tae Hwan Jung (Jeff Jung), "Simple text-generator with OpenAI GPT-2 Pytorch implementation" [GitHub repository](#), which uses code and models from [this repo](#).

Prepare the environment

Tasks (read the notebook for exact instructions)

- ▶ Generate some unconditional fake text.
- ▶ Auto-complete some text.
- ▶ Generate multiple novel text samples.
- ▶ Generate samples of varying length.
- ▶ Modify the model parameters.

Tasks (read the notebook for exact instructions)

- ▶ Generate some unconditional fake text.
- ▶ Auto-complete some text.
- ▶ Generate multiple novel text samples.
- ▶ Generate samples of varying length.
- ▶ Modify the model parameters.

Tasks (read the notebook for exact instructions)

- ▶ Generate some unconditional fake text.
- ▶ Auto-complete some text.
- ▶ Generate multiple novel text samples.
- ▶ Generate samples of varying length.
- ▶ Modify the model parameters.

Tasks (read the notebook for exact instructions)

- ▶ Generate some unconditional fake text.
- ▶ Auto-complete some text.
- ▶ Generate multiple novel text samples.
- ▶ Generate samples of varying length.
- ▶ Modify the model parameters.

Tasks (read the notebook for exact instructions)

- ▶ Generate some unconditional fake text.
- ▶ Auto-complete some text.
- ▶ Generate multiple novel text samples.
- ▶ Generate samples of varying length.
- ▶ Modify the model parameters.

Tasks (read the notebook for exact instructions)

- ▶ Generate some unconditional fake text.
- ▶ Auto-complete some text.
- ▶ Generate multiple novel text samples.
- ▶ Generate samples of varying length.
- ▶ Modify the model parameters.

How to get points?

- ▶ To get the points for this week's exercise, answer the questions in MyCourses related to this Computer Exercise B.