



RAG-time

In this project you will use a language model to create a project that will display up-to-date information from the Internet, whenever it runs.

You will use a technique called “RAG” using the same sort of artificial intelligence technology that powers tools like ChatGPT.

This project assumes you already understand the basics of how language models work.

If you haven’t already completed the “Language Models” worksheet, you will understand this project better if you do that worksheet first.

The screenshot shows the Scratch programming interface. On the left, the 'Code' palette is open, displaying the 'Language model' extension. It includes blocks for submitting prompts, clearing context, and setting initial context from a Wikipedia text about the UK singles chart. A 'when green flag clicked' hat is attached to a script that clears context, sets a variable to the Wikipedia text, and uses the language model to submit a question about the current number one single in the UK. On the right, the stage features a boombox sprite and a background of a concert stage with spotlights. A speech bubble from the boombox says: "'Golden' by Huntr/x is the current number one single in the UK. ↩". The bottom right corner shows the 'Stage' palette with a 'Backdrops' section containing two backdrops.

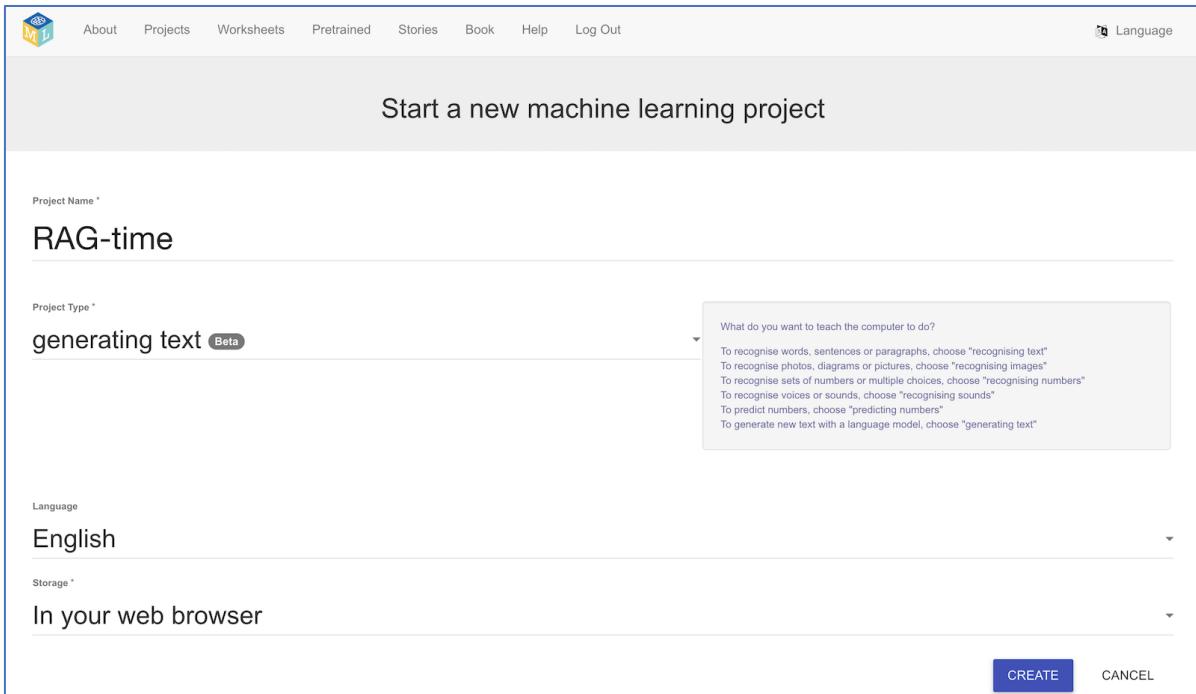


This project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License
<http://creativecommons.org/licenses/by-nc-sa/4.0/>

If you are under the age of 13, please only use a small language model with supervision from a trusted adult.

Generative AI can sometimes generate text that isn't nice or appropriate.

1. Go to <https://machinelearningforkids.co.uk/>
2. Click on “**Get started**”
3. Click on “**Log In**” and type in your username and password
If you can't remember your username or password, ask your teacher or group leader to reset it for you.
4. Click on “**Projects**” on the top menu bar
5. Click the “**+ Add a new project**” button.
6. Name your project “**RAG-time**” set it to learn how to generate text.

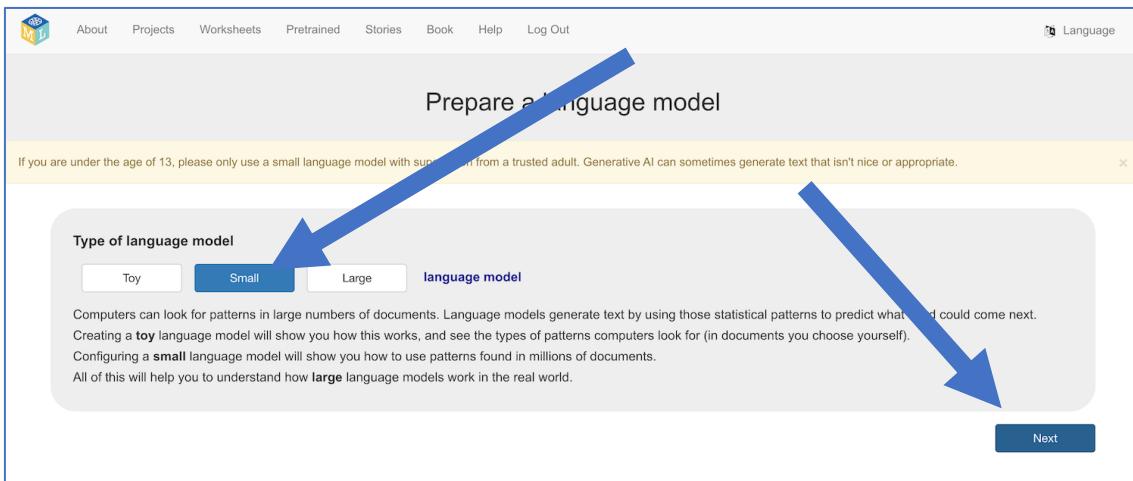


The screenshot shows a web page for creating a new machine learning project. At the top, there's a navigation bar with links for About, Projects, Worksheets, Pretrained, Stories, Book, Help, and Log Out. On the right side of the nav bar is a Language selection dropdown. Below the nav bar, the main title is "Start a new machine learning project". The first input field is "Project Name *", which contains "RAG-time". The next field is "Project Type *", with the option "generating text" selected and a "(Beta)" label. To the right of this field is a tooltip box titled "What do you want to teach the computer to do?". It lists several options: "To recognise words, sentences or paragraphs, choose "recognising text"" (which is selected), "To recognise photos, diagrams or pictures, choose "recognising images"" (disabled), "To recognise sets of numbers or multiple choices, choose "recognising numbers"" (disabled), "To recognise voices or sounds, choose "recognising sounds"" (disabled), "To predict numbers, choose "predicting numbers"" (disabled), and "To generate new text with a language model, choose "generating text"" (disabled). Below these fields is a "Language" dropdown set to "English". Further down is a "Storage *" dropdown set to "In your web browser". At the bottom right are two buttons: a blue "CREATE" button and a white "CANCEL" button.

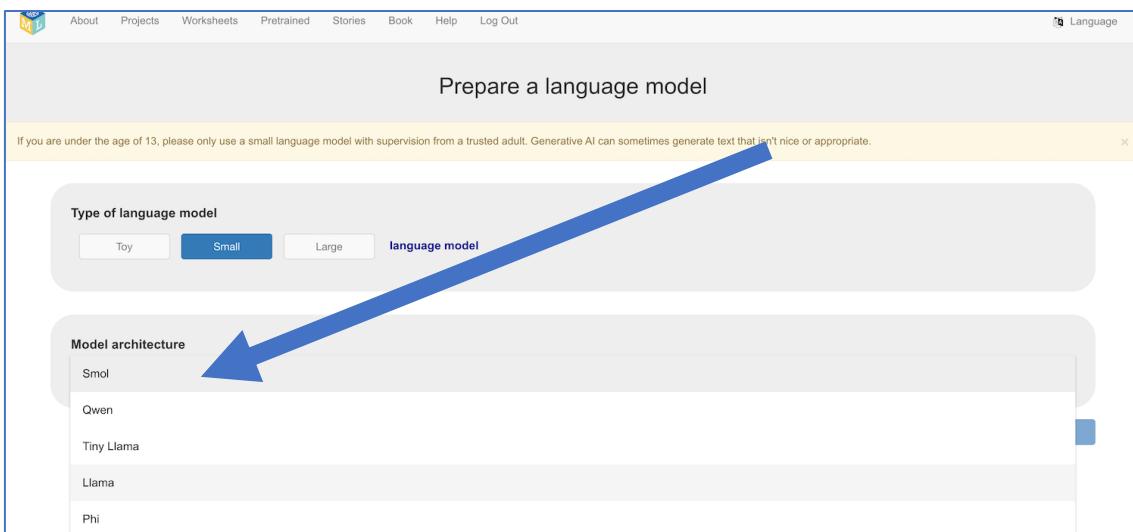
7. Click **Create**

8. You should see your new project in the projects list. Click on it.

9. Click on **Small**, and then click **Next**



10. Choose a model architecture



*This is a complex project. It will benefit from the **largest model** you can use.*

But larger models:

- * take longer to download
- * need more storage space on your computer
- * need a faster and more powerful computer to run

Ask your teacher or group leader if you are not sure how large a model you can pick.

11. Click **Download**

12. Choose a context window size

The screenshot shows a user interface for preparing a language model. At the top, there's a navigation bar with links for About, Projects, Worksheets, Pretrained, Stories, Book, Help, and Log Out. On the right, there's a Language selection dropdown. The main area is titled "Prepare a language model". A yellow banner at the top states: "If you are under the age of 13, please only use a small language model with supervision from a trusted adult. Generative AI can sometimes generate text that isn't nice or appropriate." Below this, there's a section for "Type of language model" with options Toy, Small (which is selected), Large, and language model. The "Model architecture" section shows "Gemma" as the selected option. The "Size of context window" section has an input field containing "16384", with a descriptive text explaining what it means: "How much text the model looks at before generating the next word. A smaller window means it works with less previous text. A larger window allows it to use more, which can make responses more accurate but needs more computer memory." A blue arrow points to the "16384" input field. In the bottom right corner of the main area, there's a "Next" button.

*This is a complex project. It will benefit from the **largest context window you can use**.*

But larger context windows need a faster and more powerful computer to run

Ask your teacher or group leader if you are not sure how large a window to pick.

13. Click Next

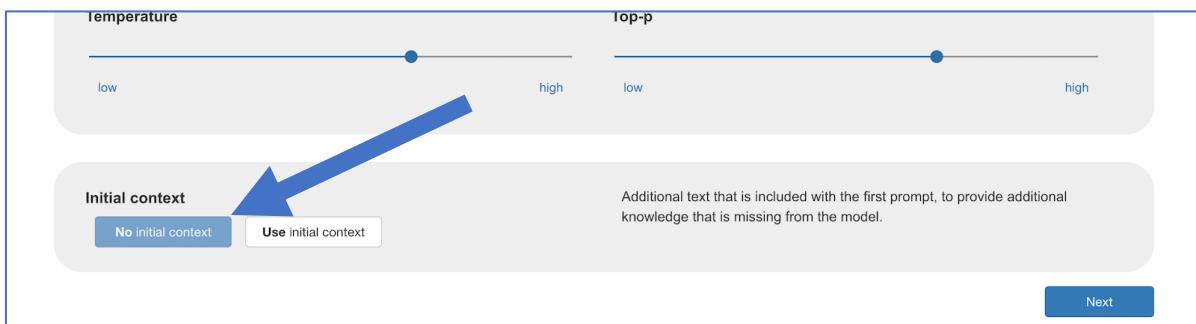
14. Set temperature and Top-p values

Values somewhere between half-way and full are a good place to start

The screenshot shows a interface for setting temperature and Top-p values. There are two sections: "Temperature" and "Top-p". Each section has a horizontal slider with a central dot. Below each slider is a description. The "Temperature" section says: "With a low temperature, the model will generate the next word based on what is most common. With a high temperature, the model is more likely to pick something unexpected." The "Top-p" section says: "With a low value, the model will only consider the most common candidates for the next word. With a high value, the model will consider any word that could come next." In the bottom right corner of the main area, there's a "Next" button.

15. Click Next

16. Select No initial context



17. Click Next

Now that you have a small language model ready to use, give it a try by putting a few simple questions in the **Prompt** box and click **Generate** to check that everything is working.

Model architecture

Gemma

Size of context window

16384

Temperature

Top-p

Initial context

No initial context Use initial context

Prompt

Generated text

Reset Generate

Review Review Review

Click on the **Review** buttons if you would like to change any of the decisions that you made.

18. Ask a general knowledge question that everyone knows
For example, “What city is the Statue of Liberty in?”

What city is the Statue of Liberty in?

The Statue of Liberty is in New York City! 🌉😊

The model will hopefully give a correct answer.

19. Click **Reset**

20. Ask a question that relies on knowledge that has never been online
For example, I asked a question about a story I wrote

What is the name of the magician in "The Vanishing Magician" by Dale Lane?

The magician's name in "The Vanishing Magician" is **Sam**. ↗😊

The model might say that it doesn't know.
Or it may “hallucinate” (make up an incorrect answer)

21. Click **Reset**

22. Ask a question that relies on knowledge of something recent
For example, I asked a question about the music charts

What is the current number one single in the UK?

That's a great question! I can't give you exact real-time info like the current number one song. ↗

To find out, you could check a website like the Official Charts Company! They'll tell you all about the charts. 😊

For example, I asked a question about election results

Who is the current Prime Minister of the United Kingdom?

The current Prime Minister of the United Kingdom is Rishi Sunak. 😊

The model might say that it doesn't know.
It may give out of date information.
Or it may “hallucinate” (make up an incorrect answer)

What is happening?

Language models generate text using the information that was used to train them.

Training these models takes months, so it cannot be done very frequently. The model you are using might have been trained up to 6 – 12 months ago. It is not possible for it to contain knowledge about things that have happened since then.

If something is private and cannot be guessed from public information, it cannot contain knowledge about those things, whenever they happened.

In this project, you will learn how we can deal with these issues.

23. Click on Review for the Initial context

The screenshot shows a user interface for a language model. At the top, there's a dropdown labeled "Model architecture" set to "Gemma". Below it is a "Size of context window" input field containing "16384" with a "Review" button to its right. There are two horizontal sliders: "Temperature" and "Top-p", both ranging from "low" to "high". Under "Temperature", the slider is positioned near the middle. Under "Top-p", it is also near the middle. At the bottom, there's a section for "Initial context" with two buttons: "No initial context" (which is selected) and "Use initial context". A large blue arrow points from the left towards the "Use initial context" button. At the very bottom, there's a "Prompt" input field, a "Reset" button, and a prominent "Generate" button.

24. Click on Use initial context

25. Collect text that contains knowledge to answer your first question.

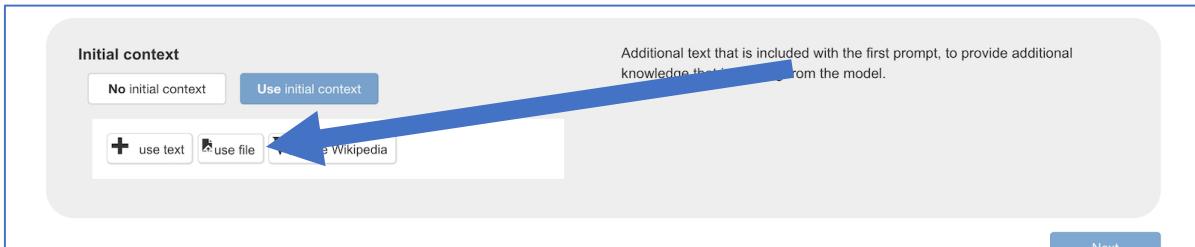
I have the story that I wrote – “The Vanishing Magician”

If you don't have anything like that, write a few lines now.

It doesn't need to be long.

26. Add your text as the initial context

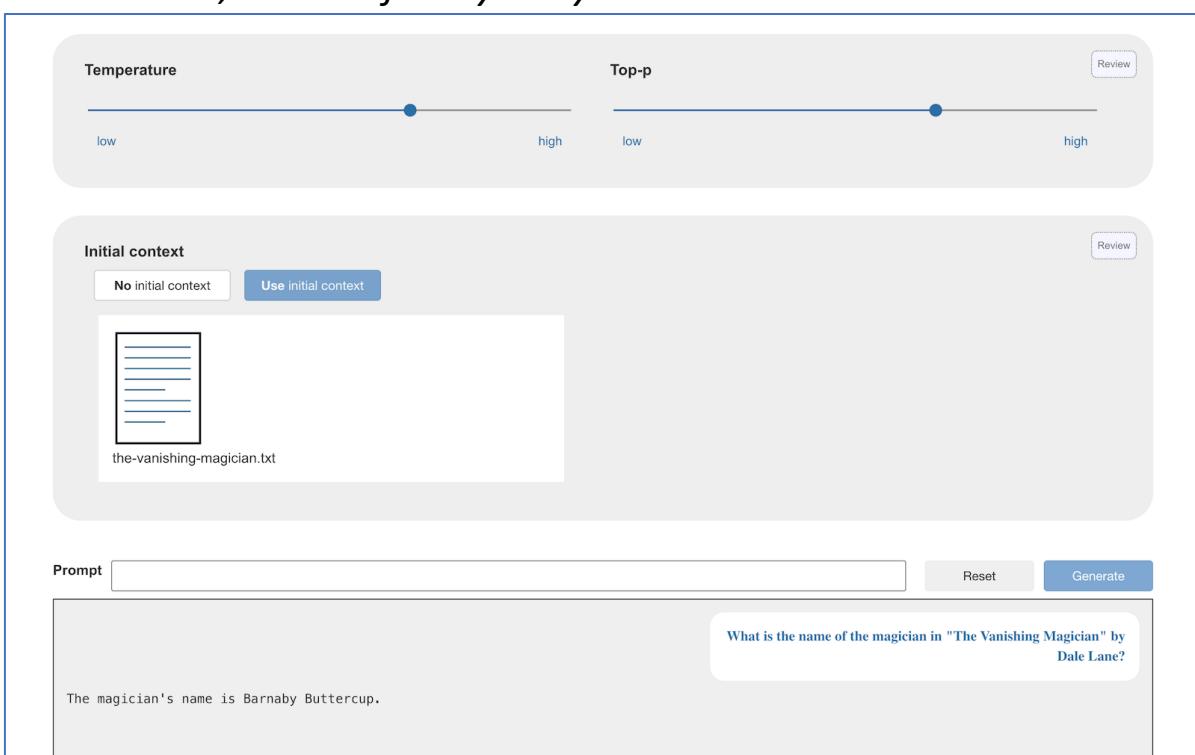
If you have it in a text (txt) file, click on **use file**
Otherwise, click on **use text** and copy/paste it in



27. Click on Next

28. Ask your first question again

I asked my first question again, about the main character of my story.
But this time, the text for my story was included in the initial context.



If the answer to the question can be found in the initial context, the model will hopefully have found it and given a correct answer.

If the model does not give you the answer you expected, click on **Reset** and try again. If it still does not give the correct answer, rephrase your question.

29. In a separate browser window, find a Wikipedia page that has the answer to your second question

*For example, the page “**UK singles chart**” has the answer to my question “What is the current number one single in the UK?”*

The screenshot shows the Wikipedia article for the "UK singles chart". The page includes a sidebar with "History" sections for Early, Official start, Gallup era, Millward Brown era, Internet era, Christmas number ones, Streaming era, 2020s, Inclusion criteria, and Broadcasts. The main content area describes the chart's evolution and its current status as the Official Singles Chart, listing top sellers and noting its comprehensive nature. A large graphic of the Official Chart logo (a blue arrow pointing upwards) is prominently displayed on the right side of the page.

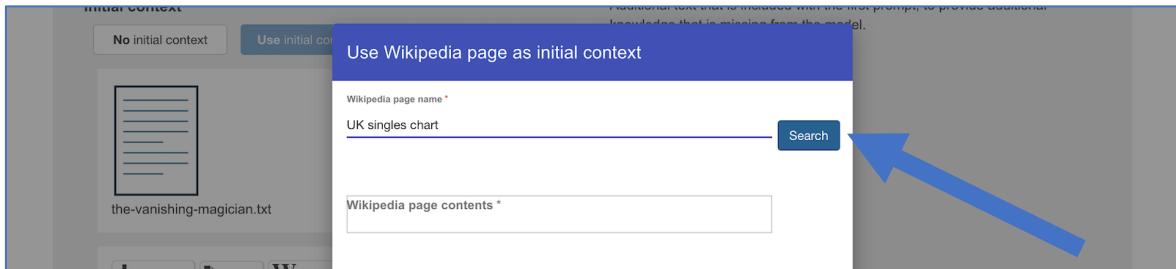
30. Click on Review for the Initial context again

The screenshot shows the AI interface's "Initial context" section. It features a "No initial context" button and a "Use initial context" button. Below these buttons is a preview window showing a document titled "the-vanishing-magician.txt". A large blue arrow points from the "Review" button at the top right towards the "Use initial context" button.

31. Click on use Wikipedia

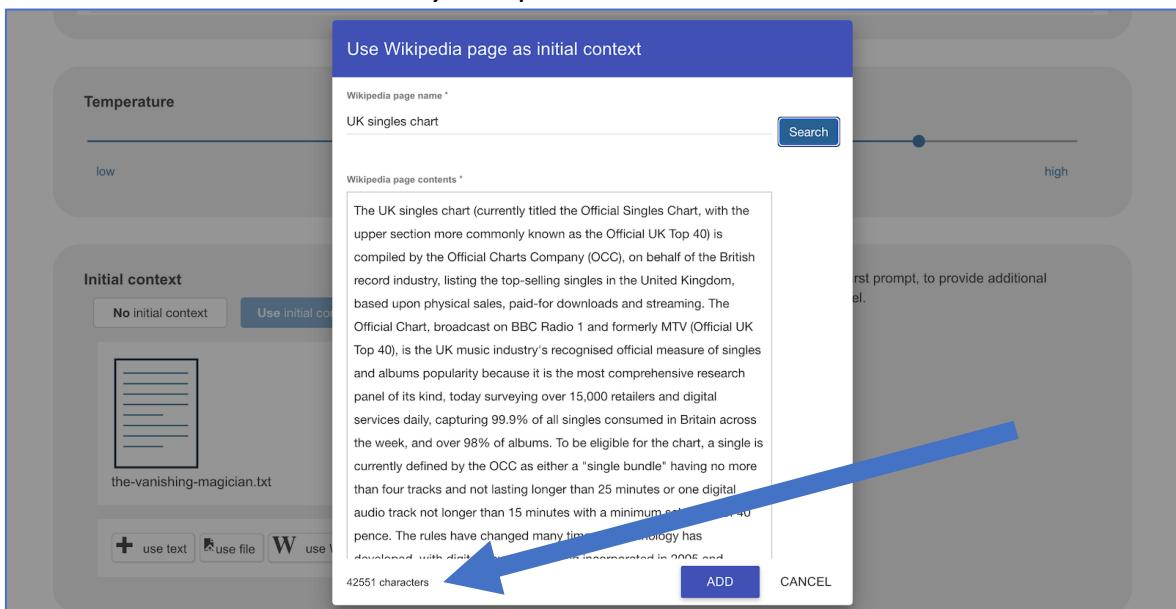
The screenshot shows the AI interface's "Initial context" section. It features a "No initial context" button and a "Use initial context" button. Below these buttons is a preview window showing a document titled "the-vanishing-magician.txt". At the bottom of the interface, there are three buttons: "+ use text", "use file", and "W use Wikipedia". A large blue arrow points from the "use Wikipedia" button at the bottom left towards the "use Wikipedia" button at the bottom right.

32. Enter the name of the Wikipedia page you found. Click on Search



33. Look at the size of the Wikipedia page

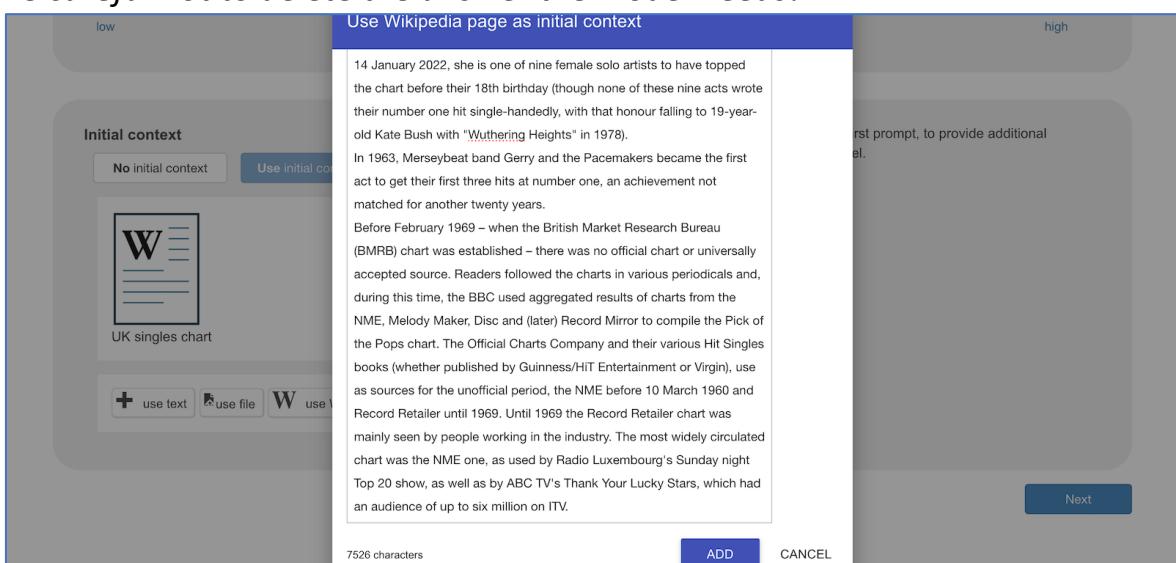
If it is very large, your computer will need a lot of memory to keep this whole page in the context while it answers your question.



34. Delete the end of the page until it is smaller than 10000 characters

Delete from the bottom of the page – keep what is at the start

Be careful not to delete the answer the model needs!



35. Make a note of approximately how many characters you kept
For the “UK singles chart” page, I kept about the first 7500 characters

36. Click on **Add**

37. Click on **Next**

38. Ask your second question again

*I asked my question about the song at the top of the UK charts again.
But this time, recent chart information was included in the initial context.*



If the answer to the question can be found in the initial context, the model will hopefully have found it and given a correct answer.

If the model does not give you the answer you expected, click on **Reset** and try again. Rephrasing your question may also help.

39. In a separate browser window, find a Wikipedia page with the answer to your last question

*For example, the page “**Prime Minister of the United Kingdom**” has the answer to “Who is the current Prime Minister of the United Kingdom?”*

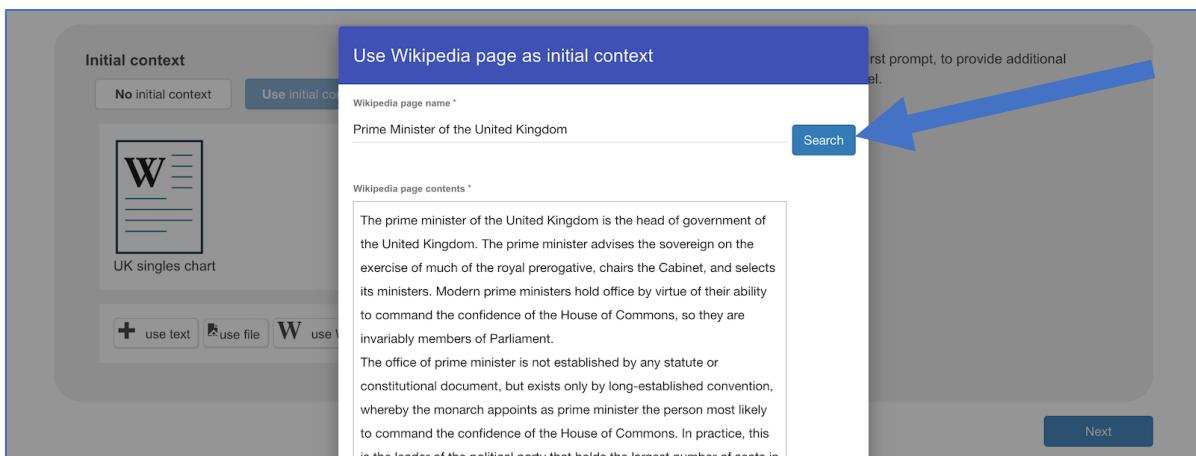
A screenshot of a Wikipedia article page for 'Prime Minister of the United Kingdom'. The page title is 'Prime Minister of the United Kingdom'. The main content area starts with a summary: 'The prime minister of the United Kingdom is the head of government of the United Kingdom. The prime minister advises the sovereign on the exercise of much of the royal prerogative, chairs the Cabinet, and selects its ministers. Modern prime ministers hold office by virtue of their ability to command the confidence of the House of Commons, so they are invariably members of Parliament.' To the right of the summary, there is a sidebar with the heading 'Prime Minister of the United Kingdom of Great Britain and Northern Ireland' and a small decorative icon.

40. Click on **Review** for the **Initial context** again

41. Click on **use Wikipedia**

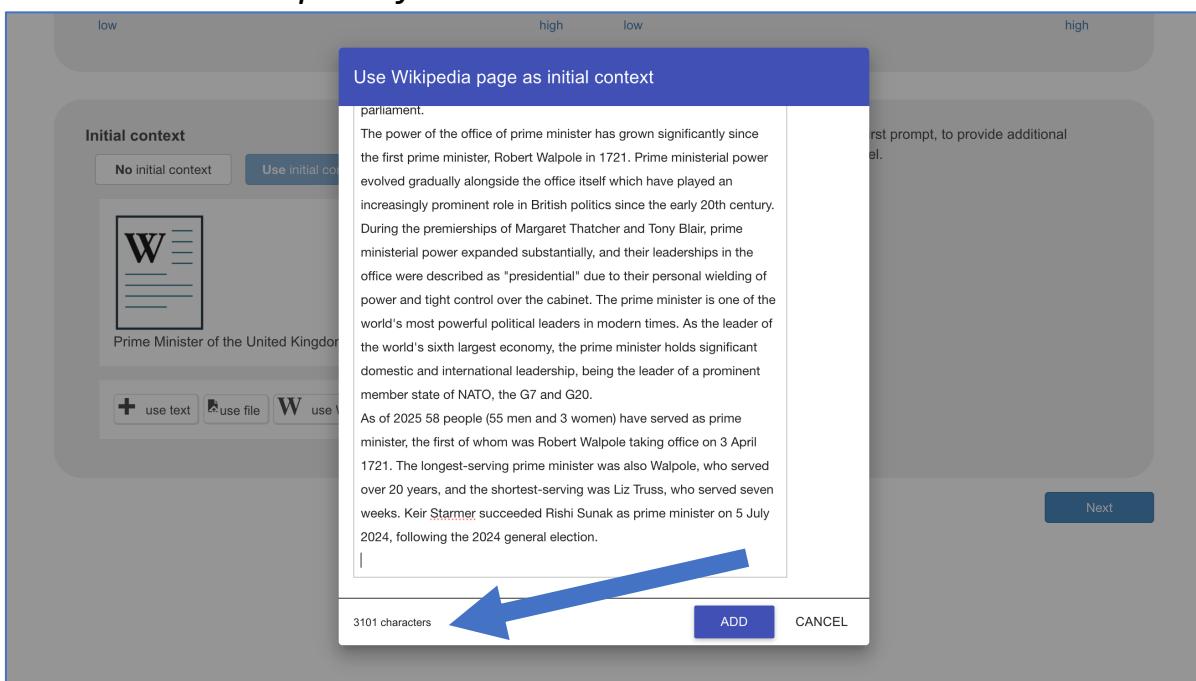
42. Enter the title for your new Wikipedia page

43. Click on **Search**



44. As before, delete **from the bottom** of the page to reduce the size of the initial context

For my Wikipedia page, keeping the first 3100 characters was enough to still include examples of the correct answer

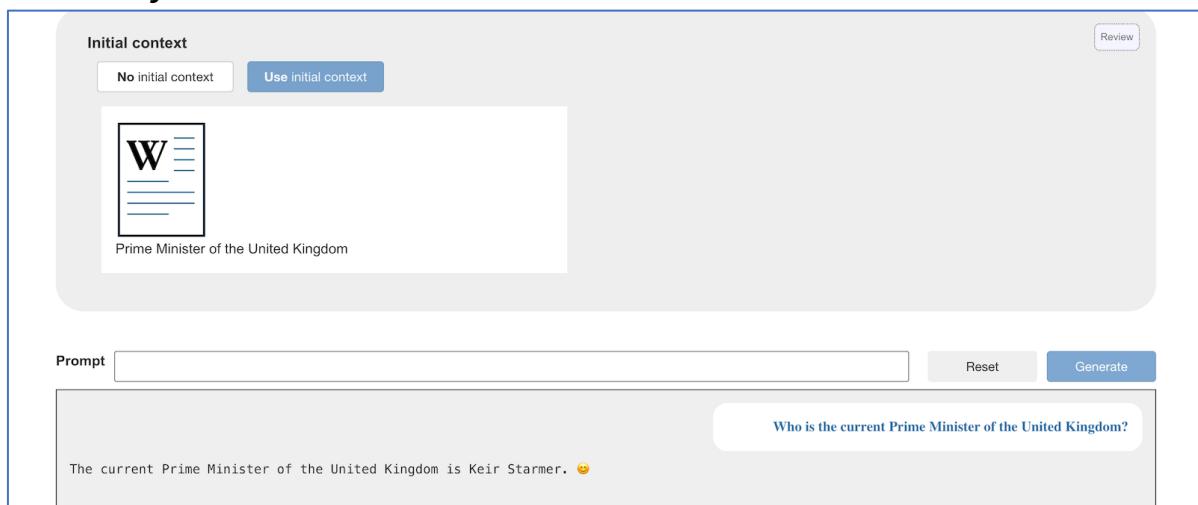


45. Click on **Add**

46. Click on **Next**

47. Ask your third question again

I asked my question about the current UK prime minister. But this time, recent information about the PM was included in the initial context.



This is called “RAG”

You have been searching for documents with additional information that the language model does not already have. You added these to the context for your question to improve the answers the model can give.

This technique is called “Retrieval Augmented Generation” because we are **retrieving** documents and using these to **augment** (improve) the answers that the language model can **generate**.

We call this **RAG** for short.

RAG projects often don’t use only one document for this – often the top three or top five search results for the question could be included in the initial context to give the model the best chance to find the answer.

You are running your RAG project on your own computer instead of a full set of servers, so you need to cut that down - but the technique is the same.

48. Click on Scratch 3

The current Prime Minister of the United Kingdom is Keir Starmer. 😊

Prompt: Who is the current Prime Minister of the United Kingdom?

Reset Generate

Scratch 3

Use your language model in Scratch

Scratch 3

49. Open the Extensions

Control

Sensing

Operators

Variables

My Blocks

Images

Language model

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

say [] for []

change y by 10

50. Add the Wikipedia extension to your project

← Back

Choose an Extension

Translate

Translate text into many languages.

Requires: WiFi

Collaboration with: Google

Wikipedia

Get the text from pages on Wikipedia.

Requires: WiFi

Collaboration with: ML for Kids

Weather

Get weather data from Open-Meteo.

Requires: WiFi

Collaboration with: ML for Kids

Books

Get book data from Open Library.

Requires: WiFi

Collaboration with: ML for Kids

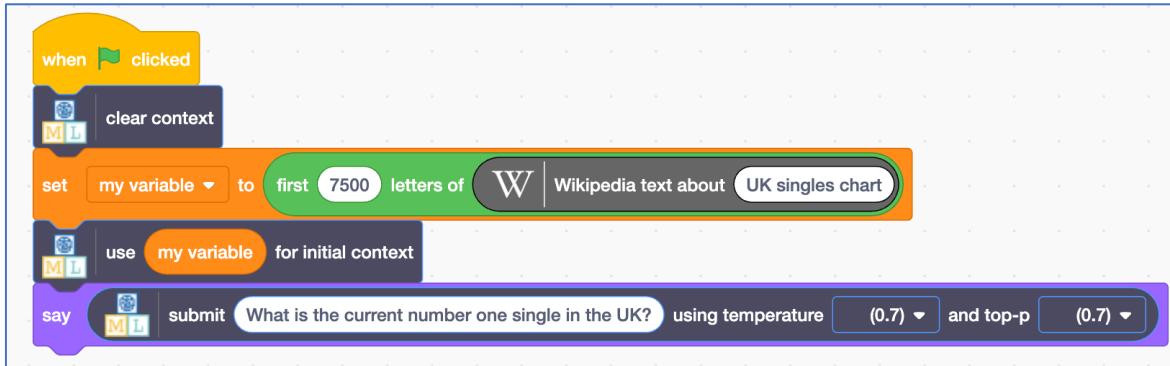
51. Create the following code

Replace these bits with what you found worked well in your testing:

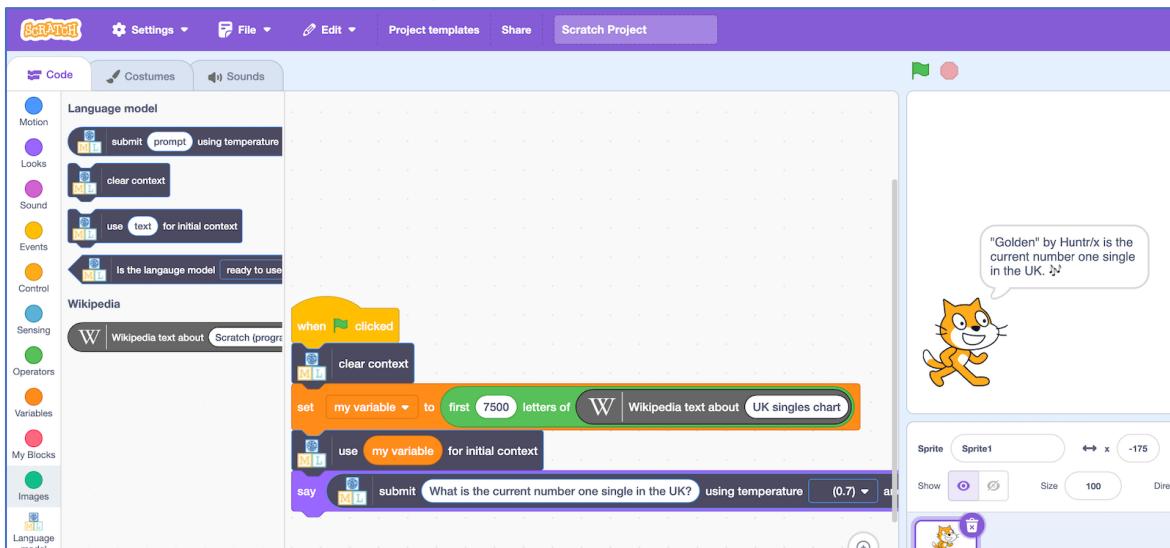
“7500” – replace this with how much of the Wikipedia page you needed to keep

“UK singles chart” – replace this with the name of the Wikipedia page you used

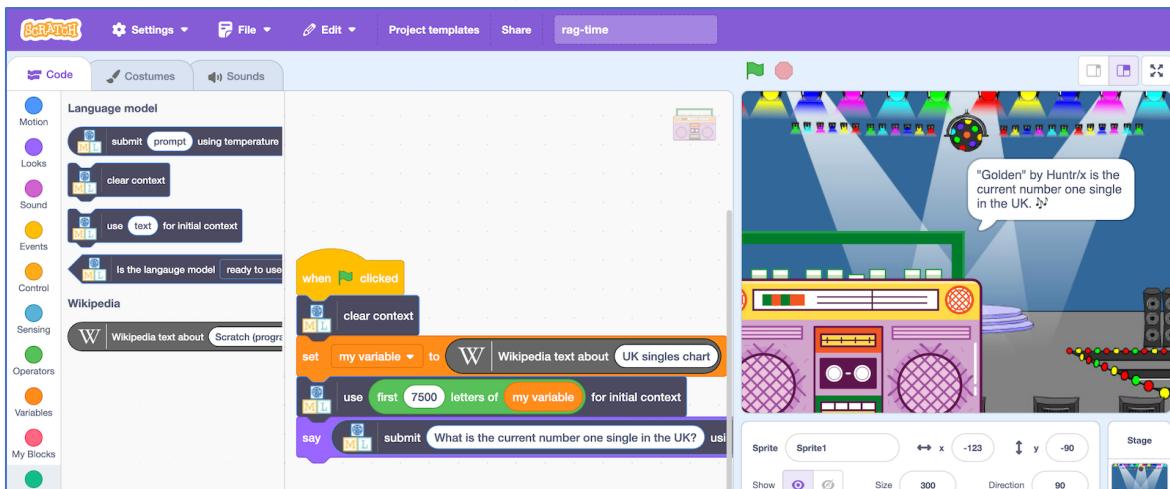
“What is the current number one single in the UK?” – replace with your question



52. Click the Green Flag



53. Try adding sprites, backgrounds, animations, or sounds



What have you done?

You've created a project that blends live text from the Internet with a language model's ability to work with text.

Live data from Wikipedia means whenever you run this project it will have fresh, current data. If you run it next week, it will use different data to what it uses today.

You are using the language model's ability to "understand" natural language to interpret the live data, and display the information you are interested in.

This is a common approach in artificial intelligence projects today, enhancing what is possible with language models by themselves.

If you include context from your private documents, you can create a system that can answer questions about your private content the model alone doesn't have any information about.

If you include context from recent documents, you can create a system that can answer questions about current events that occurred after the model was trained.

This is a common approach, often called "RAG".

A similar approach is described as "Model Context Protocol" (MCP) - which includes agreeing (a **protocol**) on the ways to provide additional **context to a model**.

These techniques are essential for creating useful systems with language models.