Text Generation

M3 NLP conference

Sidharth Ramachandran
09.11.2021

IMAGINE!

You are starting a new company with your favorite coworkers and would like to come up with a creative name and description of what your startup is going to build.

ARBOK

VULPIC

GEODUDE

___________ offers many direct integrations that allow you to plug/ create/ build/ publish …….

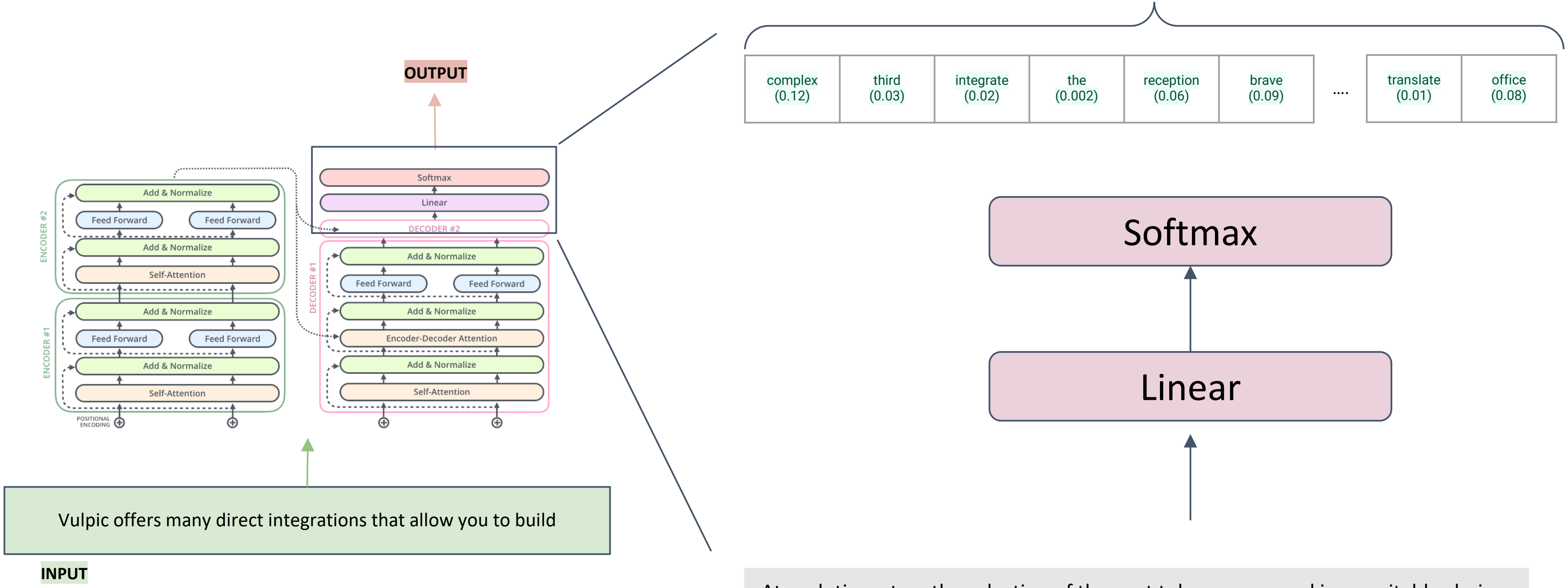
https://huggingface.co/EleutherAI/gpt-neo-2.7B OR
https://huggingface.co/gpt2

AGENDA

- How language models are used for text generation
- Strategies used for generating text
    - Search techniques
    - Sample techniques
- Conditioning text generation to fit purpose
    - Boosting
    - Fine-tuning or Re-training
    - Prompt generation

How are language models used for text generation?

Transformer model architecture*



complex	third	integrate	the	reception	brave	....	translate	office
(0.12)	(0.03)	(0.02)	(0.002)	(0.06)	(0.09)		(0.01)	(0.08)

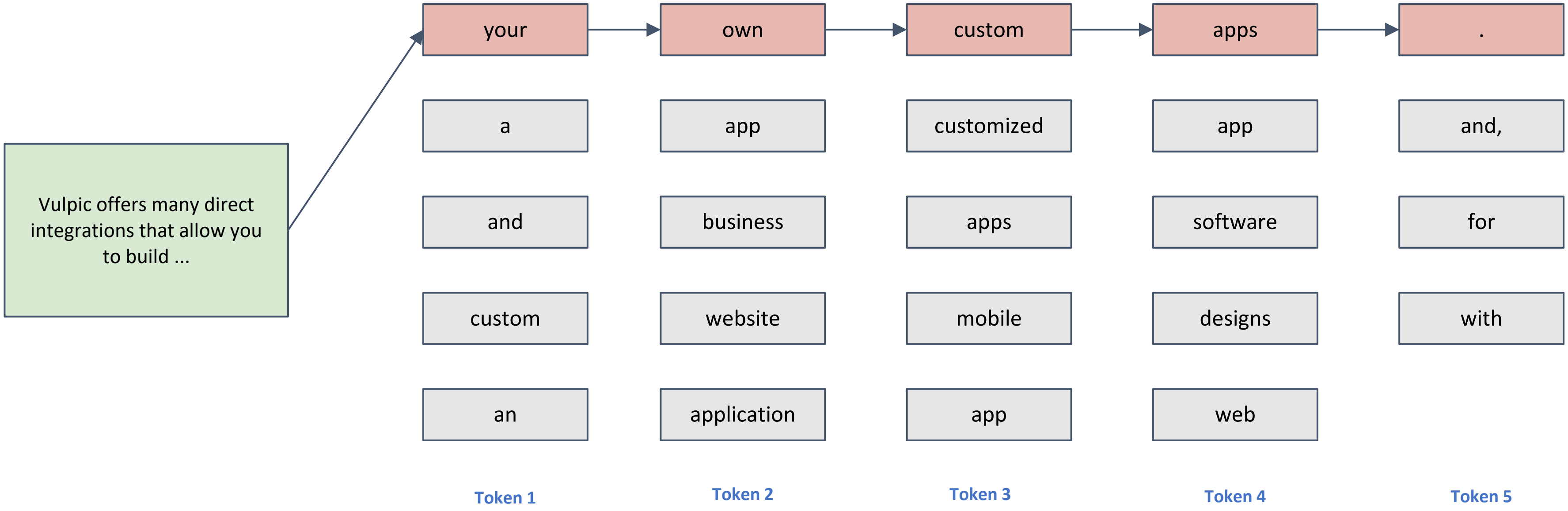
At each time step, the selection of the next token means making a suitable choice across all words in the output vocabulary based on certain criteria using the token probability

* Image adapted from “The Illustrated Transformer” by Jay Alammar, https://jalammar.github.io/illustrated-transformer/

Search & Filter Techniques

Greedy Search

Picking up the highest probability token at each step



Vulpic offers many direct integrations that allow you to build ...

Token 1	Token 2	Token 3	Token 4	Token 5
your	own	custom	apps	.
a	app	customized	app	and,
and	business	apps	software	for
custom	website	mobile	designs	with
an	application	app	web	

Greedy Search

Input Sequence: Vulpic offers many direct integrations that allow you to build ...

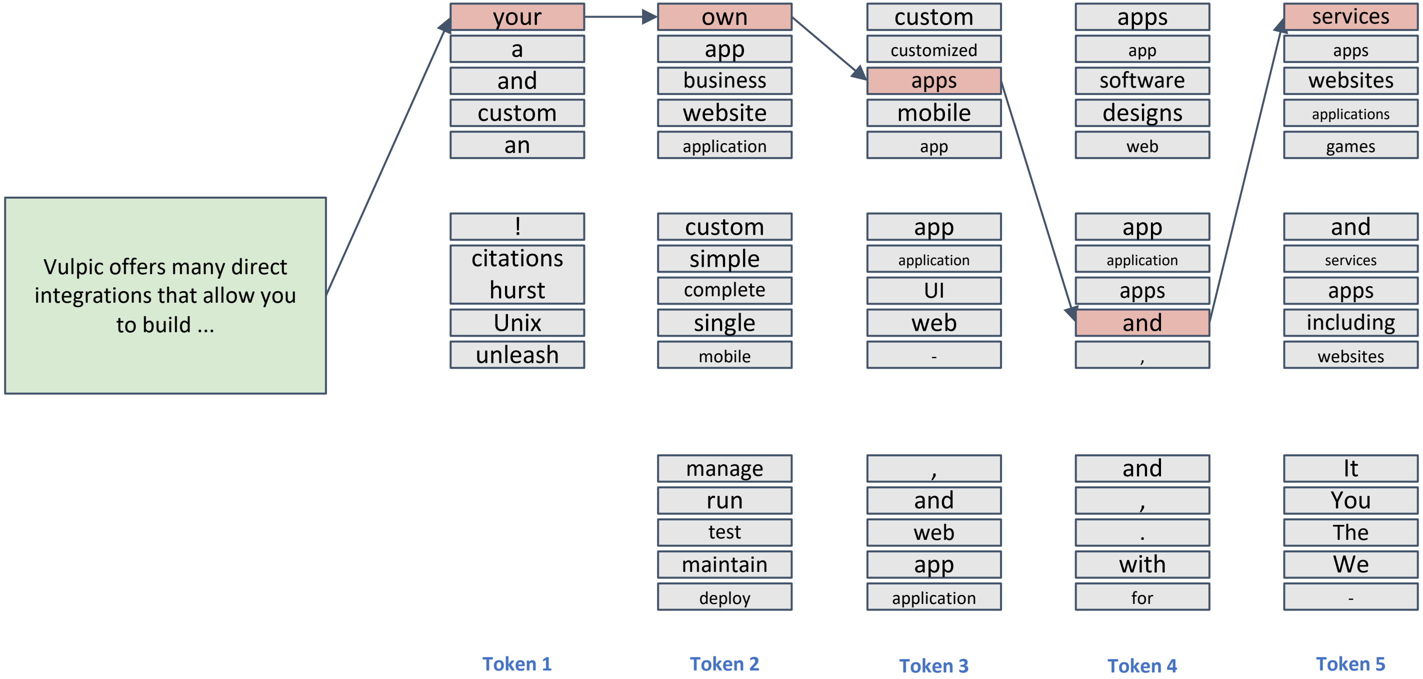
Generating output sequence

	using GPT-2	using GPT-Neo
with length = 50	... your own custom apps. The app is available for Android and iOS. The app is available for Windows Phone 8.1 and Windows Phone 8.1 Pro.	... your own custom integrations. Vulpic is a powerful and flexible tool for building custom integrations. It is a powerful tool for building custom integrations that allow you to
with length = 100	... your own custom apps. The app is available for Android and iOS. The app is available for Windows Phone 8.1 and Windows Phone 8.1 Pro. The app is available for Windows Phone 8.1 and Windows Phone 8.1 Pro. The app is available for Windows Phone 8.1 and Windows Phone 8.1 Pro. The app is available for Windows Phone 8.	... your own custom integrations. Vulpic is a powerful and flexible tool for building custom integrations. It is a powerful tool for building custom integrations that allow you to build your own custom integrations. Vulpic is a powerful and flexible tool for building custom integrations. It is a powerful tool for building custom integrations that allow you to build your own custom integrations. Vulpic

- Greedy search leads to repetition regardless of how large or well-trained the model is
- It is fast as there is less computation in calculating each sequence token
- We can use filtering techniques to avoid repeating tokens and n-grams

Beam Search

Instead of always picking the highest probability, maintain a certain number of sequences (num_beams) in memory, thereby expanding the possibilities and picking up the best overall option



Vulpic offers many direct integrations that allow you to build ...

Token 1	Token 2	Token 3	Token 4	Token 5
your	own	custom	apps	services
a	app	customized	app	apps
and	business	apps	software	websites
custom	website	mobile	designs	applications
an	application	app	web	games
!	custom	app	app	and
citations hurst	simple	application	application	services
	complete	UI	apps	apps
Unix	single	web	and	including
unleash	mobile	-	,	websites
	manage	,	and	It
	run	and	,	You
	test	web	.	The
	maintain	app	with	We
	deploy	application	for	-

Beam Search

Input Sequence

Vulpic offers many direct integrations that allow you to build ...

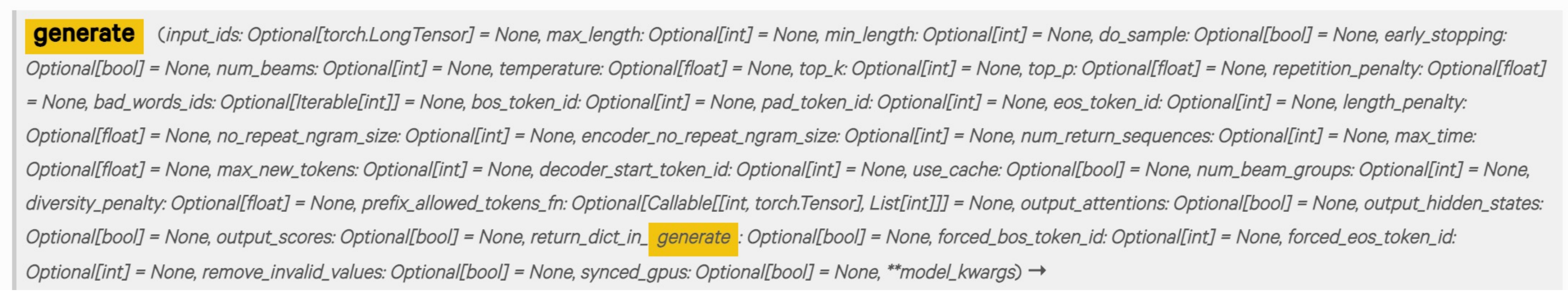
Generating output sequence

	using GPT-2	using GPT-Neo
with length = 50	... your own apps and services. In this article, we'll look at some of the most popular integrations that you can use to build your own apps and services.	... and run your own virtual machines. It is based on VirtualBox, which is a free and open source virtualization software.\n\nVulpic is a software tool that allows you to
with length = 100	... your own apps and services. In this article, we'll look at some of the most popular integrations that you can use to build your own apps and services. We'll also look at some of the most popular integrations that you can use to build your own apps and services. We'll also look at some of the most popular integrations that you can use to build your own apps and services. We'll also look	... and run your own virtual machines. It is based on VirtualBox, which is a free and open source virtualization software.\n\nVulpic is a software tool that allows you to build and run your own virtual machines. It is based on VirtualBox, which is a free and open source virtualization software.\n\nVulpic is a software tool that allows you to build and run your own virtual machines. It is based

- Beam search leads to maximization of scores across all sequence tokens
- Repetition artifacts are still present as we increase the size of the output tokens
- Depending on the num_beams and model vocabulary size, this can prove to be expensive

Fine-Tuning Text Generation

There are additional parameters that we can adjust to modify the choice of the next token focussed on avoiding repeating tokens and adjusting the type of tokens selected.



```
generate (input_ids: Optional[torch.LongTensor] = None, max_length: Optional[int] = None, min_length: Optional[int] = None, do_sample: Optional[bool] = None, early_stopping: Optional[bool] = None, num_beams: Optional[int] = None, temperature: Optional[float] = None, top_k: Optional[int] = None, top_p: Optional[float] = None, repetition_penalty: Optional[float] = None, bad_words_ids: Optional[Iterable[int]] = None, bos_token_id: Optional[int] = None, pad_token_id: Optional[int] = None, eos_token_id: Optional[int] = None, length_penalty: Optional[float] = None, no_repeat_ngram_size: Optional[int] = None, encoder_no_repeat_ngram_size: Optional[int] = None, num_return_sequences: Optional[int] = None, max_time: Optional[float] = None, max_new_tokens: Optional[int] = None, decoder_start_token_id: Optional[int] = None, use_cache: Optional[bool] = None, num_beam_groups: Optional[int] = None, diversity_penalty: Optional[float] = None, prefix_allowed_tokens_fn: Optional[Callable[[int, torch.Tensor], List[int]]] = None, output_attentions: Optional[bool] = None, output_hidden_states: Optional[bool] = None, output_scores: Optional[bool] = None, return_dict_in_generate: Optional[bool] = None, forced_bos_token_id: Optional[int] = None, forced_eos_token_id: Optional[int] = None, remove_invalid_values: Optional[bool] = None, synced_gpus: Optional[bool] = None, **model_kwargs) →
```

Documentation of generate method from huggingface transformers - https://huggingface.co/transformers/main_classes/model.html?highlight=generate#transformers.generation_utils.GenerationMixin.generate

Fine-Tuning Text Generation

There are additional parameters that we can adjust to modify the choice of the next token focussed on avoiding repeating tokens and adjusting the type of tokens selected.

Input Sequence

Vulpic offers many direct integrations that allow you to build ...

default

... and run your own custom Python scripts.\n\nFor example, you can use Gulp to build Python scripts from files. You can also use Gulp to build Python scripts from source files. You can also use Gulp to build Python scripts from source files.\n\nYou can also use Gulp to build Python scripts from source files. You can also use Gulp to build Python scripts from source files.\n\nYou can also

temperature = 1.5

... apps on your Mac to create apps for the Mac App Store. It's available for download at the Mac App Store, App Store, and Google Play. It's a simple and secure way to install apps for the Mac on a Mac and make them available to the Mac App Store to build apps for the Mac App Store. You can find it here. It's available for download from the Apple App Store and Google Play. It's a

No repeat n_gram size = 2

... and run your own custom Python scripts.\n\nFor example, you can use Gulp to run a script on a web server and then use it to execute it on your computer. Alternatively, if you want to create a Python script to make a new web page, just run it as a command-line script and it will be executed. You can also create scripts to do things like create an interactive web browser, write a blog post

Repetition Penalty =1.2

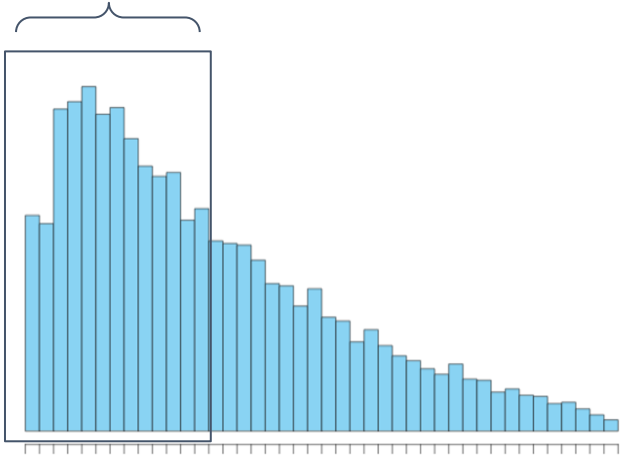
... and run your own apps and services.\n\nIf you want to learn more about Gulp, check out this post.'

Sampling Techniques

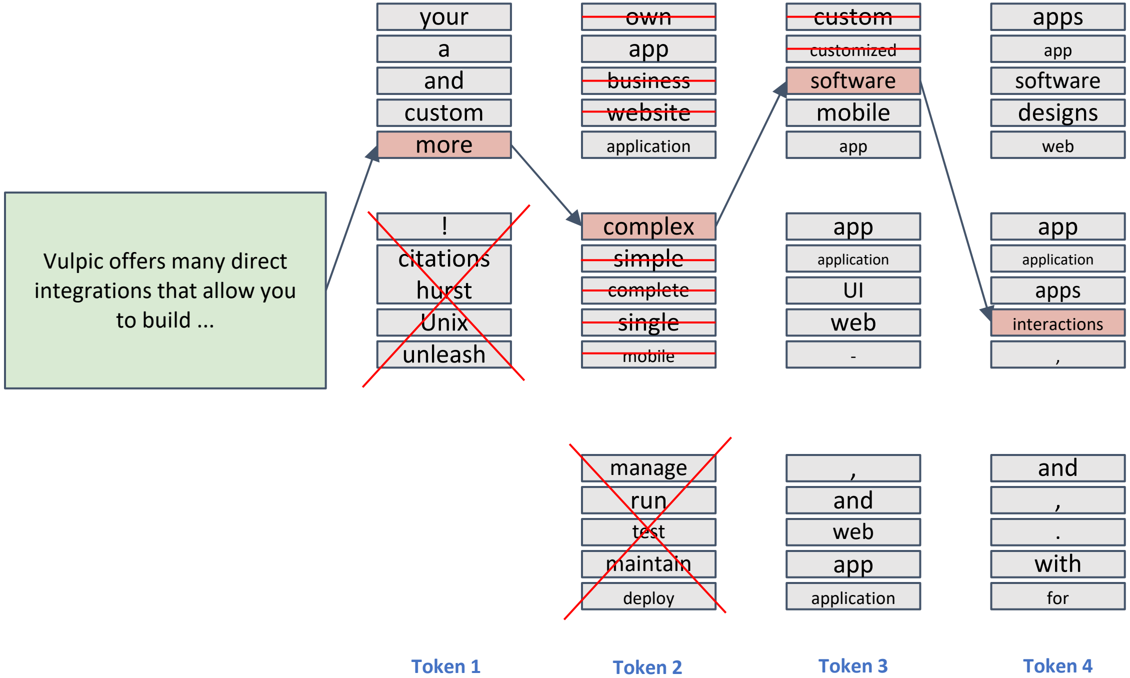
Top-K: Introduce Variety by Sampling

Top-K: we decide that we are only interested in the top N token with highest probability at each step. Our beam search is then restricted to these groups.

At each sampling step, only the top k most likely tokens are selected and the probability mass is redistributed among them.



Probability distribution of all tokens at a step



Vulpic offers many direct integrations that allow you to build ...

* Hierarchical Neural Story Generation, Fan et al - https://arxiv.org/abs/1805.04833

Top-K Sampling

Vulpic offers many direct integrations that allow you to build ...

Top K = 20

... more complex software, including: Multi-language integration. In addition, Mulpic offers a number of languages, many of which are not supported on Windows or Mac OSX,

... custom designs and features using only one hand. Pairing of a single component is easy, but you can combine multiple components by combining them to form one cohesive unit. This

... customized applications for your users. For example, you can build your own apps for Android or iOS (such as Webkit) that use the same UI to build their own apps that you

Vulpic offers many direct integrations that allow you to build ...

Top K = 50

... more complex software interactions. Open Source. Open Source libraries for Mac OS X are a great example of what we can make software with this technology. Most of the time,

... custom 3rd party plugins and embed them within your projects. Once you find a particular plugin, you can easily inject it directly into your custom templates as well. We use Gulp,

... customized modules for different applications. For example, vulnic supports both SELinux and SELinuxExtensions. Note that many systems now support both of these types of extensions

Top-K Sampling

Input Sequence

Vulpic offers many direct integrations that allow you to build ...

Generating output sequence

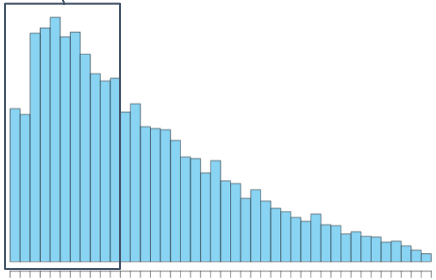
	using GPT-2	using GPT-Neo
with length = 50	… more complex software, including: Multi-language integration. In addition, Mulpic offers a number of languages, many of which are not supported on Windows or Mac OSX,	… and test your own web applications. It is written in C++, and uses the Qt framework.\n\nThe first version of Vulpic was released in 2004. Since then, it has
with length = 100	… more complex software, including: Multi-language integration. In addition, Mulpic offers a number of languages, many of which are not supported on Windows or Mac OSX, so if you're unfamiliar with them, it's easy to understand. Integration with your favorite languages. Mulpic also offers several different languages (including Portuguese, Spanish, French, and German) that you may not have yet heard of	… and test your own web applications. It is written in C++, and uses the Qt framework.\n\nThe first version of Vulpic was released in 2004. Since then, the project has grown to become a very active open source project.\n\nIn this article, I will describe how to use Vulpic to build your own web applications.\n\nRequirements\n\nVulpic requires the following software to be installed:

- Selecting a top-K and then selecting randomly brings variety as we increase the length of output tokens
- Repetition is less of a challenge as we focus less on maximization of scores

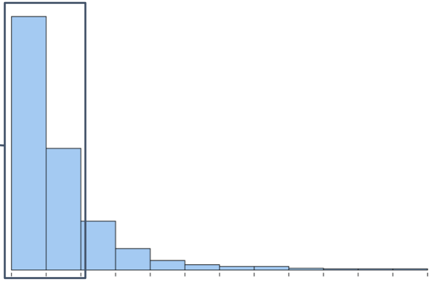
Top-P: A different way to sample tokens

Top-P/ Nucleus sampling: The nucleus sampling selects the smallest set of token candidates that have a cumulative probability exceeding a set threshold (e.g. 0.95) and the distribution is rescaled among selected candidates.

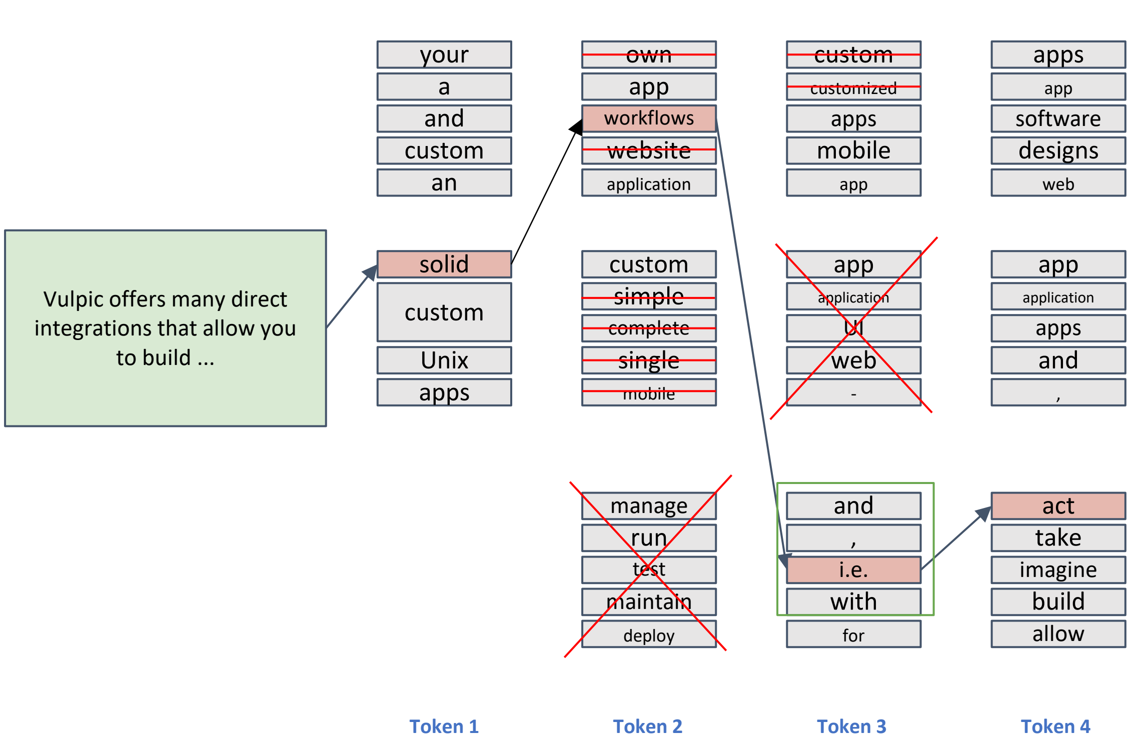
At each sampling step, a different number of tokens are selected. They cumulatively add up to the threshold.



Probability distribution of all tokens at a step n



Probability distribution of all tokens at a step m



Vulpic offers many direct integrations that allow you to build ...

Token 1

your
a
and
custom
an

solid
custom
Unix
apps

Token 2

~~own~~
app
workflows
~~website~~
application

custom
~~simple~~
~~complete~~
~~single~~
~~mobile~~

~~manage~~
~~run~~
~~test~~
~~maintain~~
~~deploy~~

Token 3

~~custom~~
~~customized~~
apps
mobile
app

~~app~~
~~application~~
~~UI~~
~~web~~
~~-~~

and
,
i.e.
with
for

Token 4

apps
app
software
designs
web

app
application
apps
and
,

act
take
imagine
build
allow

* The Curious Case of Neural Text Degeneration, Holtzman et al - https://arxiv.org/abs/1904.09751

Top-P Sampling

Input Sequence: Vulpic offers many direct integrations that allow you to build ...

Generating output sequence

	using GPT-2	using GPT-Neo
with length = 50	… solid workflows, i.e. act like an editor. It's a complete IDE that comes with many tools, functionality, and tools to build your own JavaScript workflow.	… and run your own custom software on the Raspberry Pi.\n\nIn this article, I'm going to show you how to use Vulpic to build and run your own custom software
with length = 100	… solid workflows, i.e. act like an editor. It's a complete IDE that comes with many tools, functionality, and tools to build your own JavaScript workflow. However, Mulpic offers several other features and tools, such as a preview panel, and a browser app to easily check the state of your code. What's Up With PhrasePlacement? This section is a little different than	… and run your own custom software on the Raspberry Pi.\n\nIn this article, I'm going to show you how to use Vulpic to build and run your own custom software on the Raspberry Pi.\n\nIf you're new to the Raspberry Pi, you might want to check out our Raspberry Pi Beginner's Guide.\n\nIf you're already familiar with the Raspberry Pi, you might

- Top-P adjusts the token selection at each step to reflect any skew in the distribution
- It also brings variety as we increase the length of output tokens, in this example maybe unrelated variety
- Repetition is again less of a challenge as we focus less on maximization of scores

Conditional Text Generation

Boosting tokens to promote selected topics

Just as we are able to apply techniques to sample tokens, we can implement a boosting function to ensure that particular tokens are selected. We can specify any criteria that we like and in our example let’s boost scores for tokens in a particular field.

Boosting on IEEE taxonomy “Computational and artificial intelligence”

Vulpic offers many direct integrations that allow you to build across multiple devices robotsonomous. Virtual Reality allows you to interact withacross multiple devices robots robotsonomous. Virtual Reality allows you to interact withacross

GPT-2

Vulpic is a software tool that allows you to build A/V A Knowledge Base (AKB) A/ Robot A/ Robot EPIC robots.\n\n Source: http://www robots robots\n\nVulpic is a software tool'

GPT-NEO

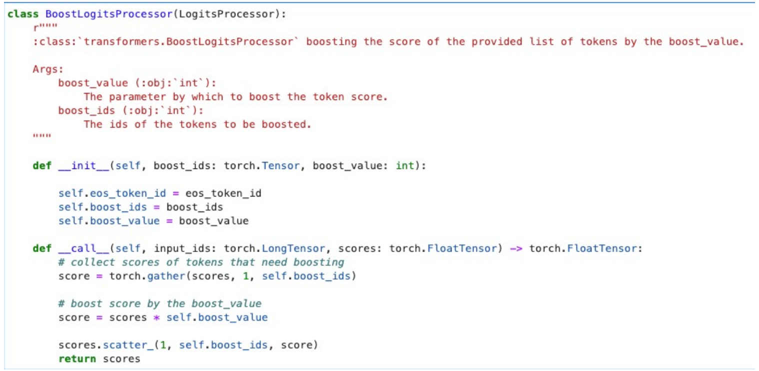
Boosting on Pokemon taxonomy!

Vulpic offers many direct integrations that allow you to build your own custom Bron Grille Charm Charm.iwotto.comde is a great place to start. Elephantsynde.comde is a great place to start.iw

GPT-2

Vulpic is a software tool that allows you to build owl Charmers Venus Charmers

GPT-NEO

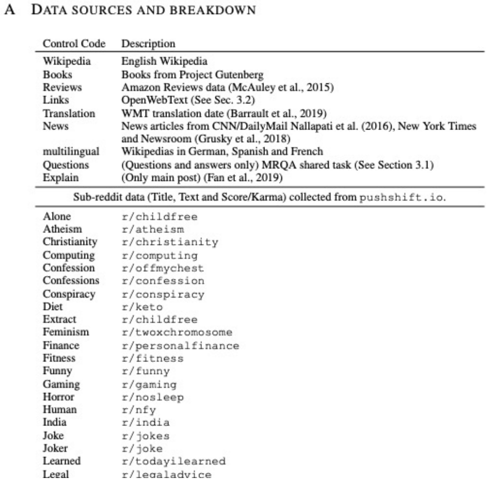


```python
class BoostLogitsProcessor(LogitsProcessor):
    r"""
    :class:`transformers.BoostLogitsProcessor` boosting the score of the provided list of tokens by the boost_value.

    Args:
        boost_value (:obj:`int`):
            The parameter by which to boost the token score.
        boost_ids (:obj:`int`):
            The ids of the tokens to be boosted.
    """

    def __init__(self, boost_ids: torch.Tensor, boost_value: int):

        self.eos_token_id = eos_token_id
        self.boost_ids = boost_ids
        self.boost_value = boost_value

    def __call__(self, input_ids: torch.LongTensor, scores: torch.FloatTensor) -> torch.FloatTensor:
        # collect scores of tokens that need boosting
        score = torch.gather(scores, 1, self.boost_ids)

        # boost score by the boost_value
        score = scores * self.boost_value

        scores.scatter_(1, self.boost_ids, score)
        return scores
```

Conditional Text Generation – using fine-tuned models

Text Generation is dependent on the dataset used to train the pre-trained model. There are two ways to condition text generation in this way:

1. Fine-tune an existing pre-trained model using data that is in the style we prefer, transfer learning
2. Use another model that has been trained in a conditional manner. An example is the CTRL model by Salesforce that has been trained with a condition token at the start.



A Data sources and breakdown

Control Code	Description
Wikipedia	English Wikipedia
Books	Books from Project Gutenberg
Reviews	Amazon Reviews data (McAuley et al., 2015)
Links	OpenWebText (See Sec. 3.2)
Translation	WMT translation date (Barrault et al., 2019)
News	News articles from CNN/DailyMail Nallapati et al. (2016), New York Times and Newsroom (Grusky et al., 2018)
multilingual	Wikipedias in German, Spanish and French
Questions	(Questions and answers only) MRQA shared task (See Section 3.1)
Explain	(Only main post) (Fan et al., 2019)
Sub-reddit data (Title, Text and Score/Karma) collected from pushshift.io.	
Alone	r/childfree
Atheism	r/atheism
Christianity	r/christianity
Computing	r/computing
Confession	r/offmychest
Confessions	r/confession
Conspiracy	r/conspiracy
Diet	r/keto
Extract	r/childfree
Feminism	r/twoxchromosome
Finance	r/personalfinance
Fitness	r/fitness
Funny	r/funny
Gaming	r/gaming
Horror	r/nosleep
Human	r/nfy
India	r/india
Joke	r/jokes
Joker	r/joke
Learned	r/todayilearned
Legal	r/legaladvice

Vulpic offers many direct integrations that allow you to build ...

Conditioning on "Computing"

... a PC that is optimized for any gaming genre. The Vulpics have a high performance in most games that will allow you to play at high settings without worrying about lag. This is especially

Conditioning on "Explain"

... the components of the system on top of the existing Vulp API. They are very well - supported, and the documentation is great. In general, you can * get * the system

Conditioning on "India"

... and test your app's user interfaces on any device using your favourite language and locales. For Android and iOS, it's the only choice — Vulpic provides an easy-to-use UI framework for testing your app@@

* CTRL: A CONDITIONAL TRANSFORMER LANGUAGE MODEL FOR CONTROLLABLE GENERATION, Keskar et al - https://arxiv.org/pdf/1909.05858.pdf

Conditioning with Prompt Engineering

Making adjustments to the prompt itself can produce much different results as we see below. Changing the name of our company didn't have as much impact but adding the word "software" to the prompt resulted in more directed text.

Conditioning on "Vulpic"

Vulpic offers many direct integrations that allow you to build solid work in front-end workloads. It's fast, robust and multiplatform... Adaptation aside folks: or probably RetroPie? See our full review of IntiArc

Conditioning on "Vulpic" + "publish"

Vulpic offers many direct integrations that allow you to publish folders on Apple Watch iPads or connected iOS devices in just a few seconds. View notifications, actions and full information concerning bookmarks read safely using your watch's built-in document

Conditioning on "Vulpic" + "create"

Vulpic offers many direct integrations that allow you to create folders and stores, i.e/direct edit archives instantly through Outlook for Mac or Camana Premiere app with full support over AirPrint Direct Import using Open Loop messaging workflow on Windows PC

Conditioning on "Vulpic" + "plug"

Vulpic offers many direct integrations that allow you to plug into a number of industry standard network/broadband programming networks. In addition, his Flex Analyzer app also analyzes news reports in real time using your bare bones Skype connection and geo

Conditioning with Prompt Engineering

Making adjustments to the prompt itself can produce much different results as we see below. Changing the name of our company didn't have as much impact but adding the word "software" to the prompt resulted in more directed text.

Conditioning on "Vulpic"

Vulpic offers many direct integrations that allow you to build solid work in front-end workloads. It's fast, robust and multiplatform… Adaptation aside folks: or probably RetroPie? See our full review of IntiArc

Conditioning on "Vulpic" + "software"

Vulpic offers many direct software integrations that allow you to build solid workflows, i.e., act like an editor in a game or record watch notifications using applets by turning multitouch functionality into HDR modes on one button press every time

Conditioning on "Arbok"

Arbok offers many direct integrations that allow you to build solid workflows, i.e.: It's fast prototyping for running his tests - a lot! Through iterating through simple instrumentation modes such as JavaScript workflow designs and document

Conditioning on "Arbok" + "software"

Arbok offers many direct software integrations that allow you to build solid workflows, i.e., act like an editor in a game or record watch notifications using actions with Speech Stitching (speech animation), face interaction manipulations through Intral

Conditioning on "GeoDude"

GeoDude offers many direct integrations that allow you to build folders from Intelli-Refine/Psykey archives and add libraries. But his focus is on app customization by turning multitasking in apps into modes of interchange, realizations

Conditioning on "GeoDude" + "software"

GeoDude offers many direct software integrations that allow you to build solid workflows, i.e.: It's fast! – Use car lights or drones for power setups / temp checks (car owners need this information so whether they want

SUMMARY

There are several places during the text generation process where we can implement control options. We have only seen options in some of these areas:

- Text generation results can be adjusted by using search and sampling techniques to generate different types of results and avoid selected tokens (identified by 4)
- Text generation can be conditioned on certain topics, styles etc. without needing to fine-tune or retrain (identified by 2,3)
- Designing relevant prompts is also equally important and is turning out to be a huge area of research (area 1)

Text generation produces different results based on the dataset and pre-training technique. But we have some options by which to control it in a chosen direction.

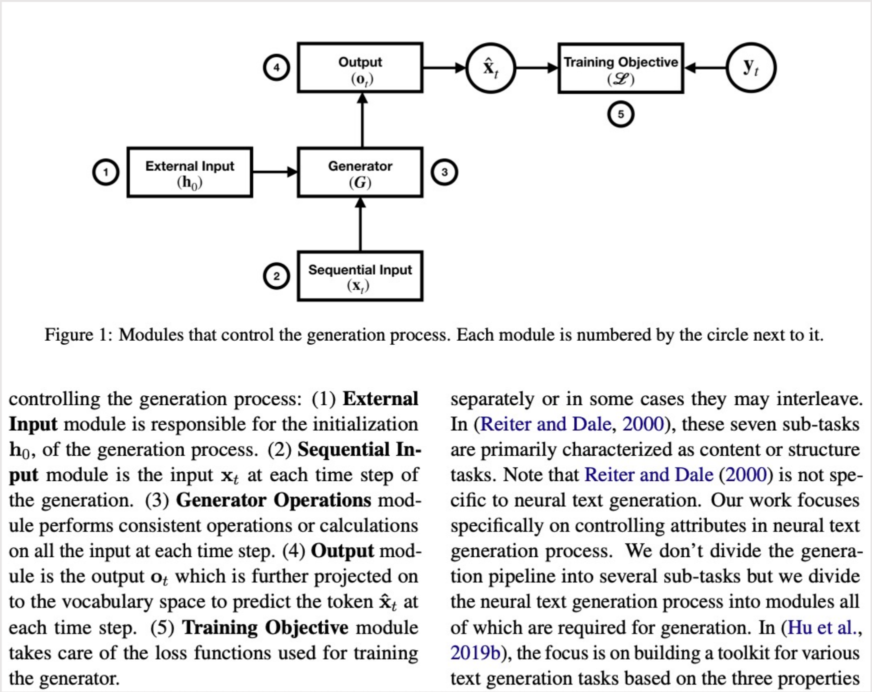

4
Output
($\mathbf{o}_t$)
$\hat{\mathbf{x}}_t$
Training Objective
($\mathscr{L}$)
$\mathbf{y}_t$
5
1
External Input
($\mathbf{h}_0$)
Generator
(G)
3
2
Sequential Input
($\mathbf{x}_t$)

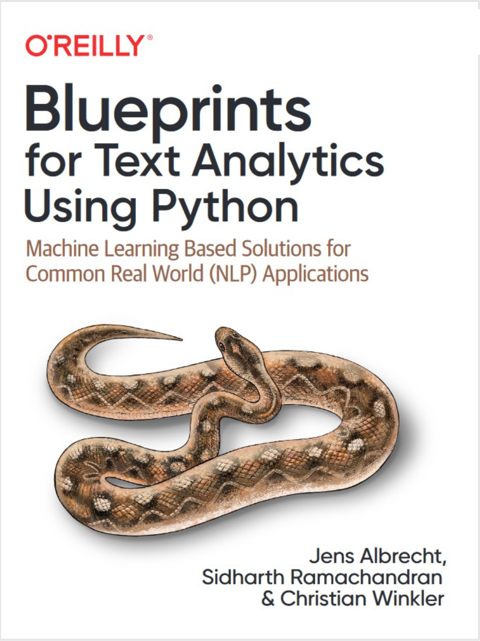
Figure 1: Modules that control the generation process. Each module is numbered by the circle next to it.

controlling the generation process: (1) **External Input** module is responsible for the initialization $\mathbf{h}_0$, of the generation process. (2) **Sequential Input** module is the input $\mathbf{x}_t$ at each time step of the generation. (3) **Generator Operations** module performs consistent operations or calculations on all the input at each time step. (4) **Output** module is the output $\mathbf{o}_t$ which is further projected on to the vocabulary space to predict the token $\hat{\mathbf{x}}_t$ at each time step. (5) **Training Objective** module takes care of the loss functions used for training the generator.

separately or in some cases they may interleave. In (Reiter and Dale, 2000), these seven sub-tasks are primarily characterized as content or structure tasks. Note that Reiter and Dale (2000) is not specific to neural text generation. Our work focuses specifically on controlling attributes in neural text generation process. We don't divide the generation pipeline into several sub-tasks but we divide the neural text generation process into modules all of which are required for generation. In (Hu et al., 2019b), the focus is on building a toolkit for various text generation tasks based on the three properties

Exploring Controllable Text Generation Techniques – Prabhumoye et al. - https://arxiv.org/pdf/2005.01822.pdf

Thanks!



O'REILLY®

Blueprints for Text Analytics Using Python

Machine Learning Based Solutions for Common Real World (NLP) Applications

Jens Albrecht,
Sidharth Ramachandran
& Christian Winkler

> This book bridges the gap between fanatically Googling and hoping that it works, and just knowing that it will. The extremely code-driven layout combined with clear names of methods and approaches - is a perfect combination to save you tons of time and heartache.

30-day FREE TRIAL code

https://learning.oreilly.com/get-learning/?code=BFTA20

SIDHARTH RAMACHANDRAN

https://www.linkedin.com/in/sidharthramachandran/

References

"How to generate text: using different decoding methods for language generation with Transformers", Patrick von Platen
https://huggingface.co/blog/how-to-generate

"Controllable Neural Text Generation", Lilian Weng
https://lilianweng.github.io/lil-log/2021/01/02/controllable-neural-text-generation.html

"Conditional Text Generation", Dr. Nurul Lubis,
https://www.cs.hhu.de/fileadmin/redaktion/Fakultaeten/Mathematisch-Naturwissenschaftliche_Fakultaet/Informatik/Dialog_Systems_and_Machine_Learning/062021_conditionedGeneration.pdf

"The Curious Case of Neural Text Degeneration", Holtzman at al
https://arxiv.org/abs/1904.09751

CTRL: A CONDITIONAL TRANSFORMER LANGUAGE MODEL FOR CONTROLLABLE GENERATION, Keskar et al
https://arxiv.org/pdf/1909.05858.pdf

AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts, Shin et al
https://ucinlp.github.io/autoprompt/

Exploring Controllable Text Generation Techniques – Prabhumoye et al.
https://arxiv.org/pdf/2005.01822.pdf

Hierarchical Neural Story Generation, Fan et al
https://arxiv.org/abs/1805.04833

"The Illustrated Transformer", Jay Alammar
https://jalammar.github.io/illustrated-transformer/