CTRL: A Conditional Transformer Language Model for Controllable Generation

Nitish Shirish Keskar[*] **Bryan McCann**[*] **Lav R. Varshney, Caiming Xiong, Richard Socher**
Salesforce Research[†]

Abstract

Large-scale language models show promising text generation capabilities, but users cannot easily control particular aspects of the generated text. We release CTRL, a 1.63 billion-parameter conditional transformer language model, trained to condition on control codes that govern style, content, and task-specific behavior. Control codes were derived from structure that naturally co-occurs with raw text, preserving the advantages of unsupervised learning while providing more explicit control over text generation. These codes also allow CTRL to predict which parts of the training data are most likely given a sequence. This provides a potential method for analyzing large amounts of data via model-based source attribution. We have released multiple full-sized, pretrained versions of CTRL at `https://github.com/salesforce/ctrl`.

1 Introduction

With enough data, model capacity, and compute, generative models can learn distributions powerful enough to produce high-quality samples from complex domains. In computer vision, the advent of generative adversarial networks (Goodfellow et al., 2014) improved image generation. Much research then focused on methods for controlling the generation process and improving estimation of generative distributions (Arjovsky et al., 2017; Chen et al., 2016; Kingma & Welling, 2013).

In natural language processing, language models are often trained as conditional language models for specific tasks that require text generation (Brants et al., 2007; Sutskever et al., 2014; Rush et al., 2015). They are also used as a means of learning word vectors (Mikolov et al., 2013), document vectors (Kiros et al., 2015), or contextualized word vectors (McCann et al., 2017; Peters et al., 2018; Devlin et al., 2018) for transfer learning. The language models themselves have been transferred to new tasks through fine-tuning as well (Dai & Le, 2015; Radford et al., 2018; Howard & Ruder, 2018). Less is understood about generation that is not constrained to any specific task. Typically prompts generated by models (Fan et al., 2018) or written by humans can only be used to provide a rough guide or starting point for the generated text. This raises the question of how text generation can be controlled more explicitly.

Inspired by the degree of control available in image generation as well as the recent progress in text generation (Radford et al., 2019) and multitask learning McCann et al. (2018), we train a language model that is conditioned on a variety of control codes (Pfaff, 1979; Poplack, 1980) that make desired features of generated text more explicit. With 1.63 billion parameters, our Conditional Transformer Language (CTRL) model can generate text conditioned on control codes that specify domain, style, topics, dates, entities, relationships between entities, plot points, and task-related behavior. To preserve the generality of the language model trained in an unsupervised setting, we train CTRL on control codes derived from structure that naturally co-occurs with the raw text typically collected for training large language models. For example, large resources like Wikipedia, Project Gutenberg, and Amazon Reviews can each be assigned a domain-related control code. Smaller resources, like the content extracted from individual subreddits, often occur with both a broader domain name, `reddit`, as well as subdomain information, `r/subdomain`. In the vast majority of cases, text collected for training is associated with a URL, which often contains information pertinent to the

[*] Equal contribution.

[†] Contact: `ctrl-monitoring@salesforce.com`

text it represents. Humans can use these codes to trigger generation of text from different linguistic communities without having to understand how to prompt with particular linguistic patterns. Text can be generated in more predictable ways by controlling for content or changing the domain even when the initial prompt remains fixed.

Because all control codes can be traced back to a particular subset of the training data, CTRL can be used to predict the subset of training data that is most likely given a sequence. This explicit relationship between CTRL and its training data can be exploited to analyze the correlations that the language model has learned from each domain, and it provides a means of studying large amounts of text through the language model.

These control codes also allow for the straightforward inclusion of task-specific data in a way that improves important skills without harming the generality of the model. Control codes for question answering and machine translation make these skills easily accessible with CTRL. These codes can be combined with codes during generation to create novel cross-over between control codes that are task-specific behavior and those that are related to domain and content.

In order to push towards more controllable, general models for natural language processing, we have released multiple full-sized, pretrained versions of CTRL at `https://github.com/salesforce/ctrl`. We hope that the release leads to further research into how controllable generation can enhance natural language understanding.

2 Language Modeling

Given example sequences of the form $x = (x_1, \dots, x_n)$ where each x_i comes from a fixed set of symbols, the goal of language modeling is to learn $p(x)$. Because x is a sequence, it is natural to factorize this distribution using the chain rule of probability (Bengio et al., 2003):

$$p(x) = \prod_{i=1}^{n} p(x_i | x_{<i})$$

This decomposes language modeling into next-word prediction. Current state-of-the-art methods (Dai et al., 2019; Radford et al., 2019) train a neural network with parameters θ to minimize the negative log-likelihood over a dataset $D = \{x^1, \dots, x^{|D|}\}$:

$$\mathcal{L}(D) = -\sum_{k=1}^{|D|} \log p_\theta(x_i^k | x_{<i}^k)$$

Because language models learn $p_\theta(x_i|x_{<i})$, a new $\tilde{x}$ of length m can be generated by sequentially sampling its constituent symbols: $p_\theta(x_0), p_\theta(x_1|\tilde{x}_0), \dots, p_\theta(x_m|\tilde{x}_{<m})$.

3 Language Modeling with CTRL

CTRL is a conditional language model that is always conditioned on a control code c and learns the distribution $p(x|c)$. The distribution can still be decomposed using the chain rule of probability and trained with a loss that takes the control code into account.

$$p(x|c) = \prod_{i=1}^{n} p(x_i | x_{<i}, c) \qquad \mathcal{L}(D) = -\sum_{k=1}^{|D|} \log p_\theta(x_i^k | x_{<i}^k, c^k)$$

The control code c provides a point of control over the generation process. This is true even when sampling x_0, in contrast to the traditional language modeling framework described in Sec. 2.

CTRL learns $p_\theta(x_i|x_{<i}, c)$ by training on sequences of raw text prepended with control codes. After minimal preprocessing (described in Sec. 3.2), a single example sequence containing n tokens is embedded as a sequence of n corresponding vectors in $\mathbb{R}^d$. Each vector is the sum of a learned

token embedding and a sinusoidal positional embedding as in the original Transformer architecture (Vaswani et al., 2017). This sequence of vectors is stacked into a matrix $X_0 \in \mathbb{R}^{n\times d}$ so that it can be processed by l attention layers (Vaswani et al., 2017). The ith layer consists of two blocks, each of which preserves the model dimension d.

The core of the first block is multi-head attention with k heads that uses a causal mask to preclude attending to future tokens:

$$\begin{aligned}\text{Attention}(X, Y, Z) &= \text{softmax}\left(\frac{\text{mask}(XY^\top)}{\sqrt{d}}\right)Z \\ \text{MultiHead}(X, k) &= [h_1; \cdots ; h_k]W_o \\ \text{where } h_j &= \text{Attention}(XW_j^1, XW_j^2, XW_j^3)\end{aligned}$$

The core of the second block is a feedforward network with ReLU activation (Nair & Hinton, 2010) that projects inputs to an inner dimension f, with parameters $U \in \mathbb{R}^{d\times f}$ and $V \in \mathbb{R}^{f\times d}$:

$$FF(X) = \max(0, XU)V$$

Each block precedes core functionality with layer normalization (Ba et al., 2016; Child et al., 2019) and follows it with a residual connection (He et al., 2016). Together, they yield X_{i+1}:

$$\begin{array}{cc} \underline{\text{Block 1}} & \underline{\text{Block 2}} \\ \\ \bar{X}_i = \text{LayerNorm}(X_i) & \bar{H}_i = \text{LayerNorm}(H_i) \\ H_i = \text{MultiHead}(\bar{X}_i) + \bar{X}_i & X_{i+1} = \text{FF}(\bar{H}_i) + \bar{H}_i \end{array}$$

Scores for each token in the vocabulary are computed from the output of the last layer:

$$\text{Scores}(X_0) = \text{LayerNorm}(X_l)W_{vocab}$$

During training, these scores are the inputs of a cross-entropy loss function. During generation, the scores corresponding to the final token are normalized with a softmax, yielding a distribution for sampling a new token.

3.1 Data

We train on 140 GB of text drawing from a wide variety of domains: Wikipedia (En, De, Es, Fr), Project Gutenberg[1], submissions from 45 subreddits, OpenWebText[2], a large collection of news data (Hermann et al., 2015; Barrault et al., 2019; Sandhaus, 2008; Grusky et al., 2018), Amazon Reviews (McAuley et al., 2015), Europarl and UN data from WMT (En-De, En-Es, En-Fr) (Barrault et al., 2019), question-answer pairs (no context documents) from ELI5 (Fan et al., 2019) and the MRQA shared task[3], which includes the Stanford Question Answering Dataset (Rajpurkar et al., 2016), NewsQA (Trischler et al., 2016), TriviaQA (Joshi et al., 2017), SearchQA (Dunn et al., 2017), HotpotQA (Yang et al., 2018), and Natural Questions (Kwiatkowski et al., 2019). A full account of training data and associated control codes can be found in Table 7 in the Appendix.

3.2 Experimental Settings

We learn BPE (Sennrich et al., 2015) codes and tokenize the data using fastBPE[4], but we use a large vocabulary of roughly 250K tokens. This includes the sub-word tokens necessary to mitigate problems with rare words, but it also reduces the average number of tokens required to generate long text by including most common words. We use English Wikipedia and a 5% split of our collected OpenWebText data for learning BPE codes. We also introduce an `unknown` token so that during

[1] We use a modified version of `https://github.com/chiphuyen/lazynlp`

[2] We use a modified version of `https://github.com/jcpeterson/openwebtext.git`

[3] `https://github.com/mrqa/MRQA-Shared-Task-2019`

[4] `https://github.com/glample/fastBPE`

preprocessing we can filter out sequences that contain more than 2 unknown tokens. This, along with the compressed storage for efficient training (TFRecords) (Abadi et al., 2016), reduces our training data to 140 GB from the total 180 GB collected. Data was treated as a single stream of tokens with non-domain control codes inserted where appropriate (often at document boundaries). The stream was chunked into contiguous sequences of tokens. Each sequence originated from a domain, and it has the corresponding domain control code prepended as the first token in the sequence. In this way, domain control codes receive special treatment (Kobus et al., 2016). They are propagated to all text in the domain as the first token. This is similar to how codes and natural language sequences have been used in multi-task settings (Wu et al., 2016; Johnson et al., 2017; McCann et al., 2018) to control conditional language models. All other control codes are injected into the data without such special treatment (Moryossef et al., 2019; Caswell et al., 2019). We experimented with sequence lengths of 256 and 512 due to memory and optimization constraints. Despite training on relatively short sequences compared to other approaches, we found that a sliding-window approach allows for generation beyond these windows, and we also found little difference in quality between the two models within the first 256 tokens. Further, we note that our vocabulary is approximately 4 times larger than similar approaches, hence the effective sequence length in characters is comparable.

CTRL has model dimension $d = 1280$, inner dimension $f = 8192$, 48 layers, and 16 heads per layer. Dropout with probability 0.1 follows the residual connections in each layer. Token embeddings were tied with the final output embedding layer (Inan et al., 2016; Press & Wolf, 2016).

CTRL was implemented in TensorFlow (Abadi et al., 2016) and trained with a global batch size of 1024 distributed across 256 cores of a Cloud TPU v3 Pod for 800k iterations. Training took approximately 2 weeks using Adagrad (Duchi et al., 2011) with a linear warmup from 0 to 0.05 over 25k steps. The norm of gradients were clipped to 0.25 as in (Merity et al., 2017). Learning rate decay was not necessary due to the monotonic nature of the Adagrad accumulator. We compared to the Adam optimizer (Kingma & Ba, 2014) while training smaller models, but we noticed comparable convergence rates and significant memory savings with Adagrad. We also experimented with explicit memory-saving optimizers including SM3 (Anil et al., 2019), Adafactor (Shazeer & Stern, 2018), and NovoGrad (Ginsburg et al., 2019) with mixed results.

4 Controllable Generation

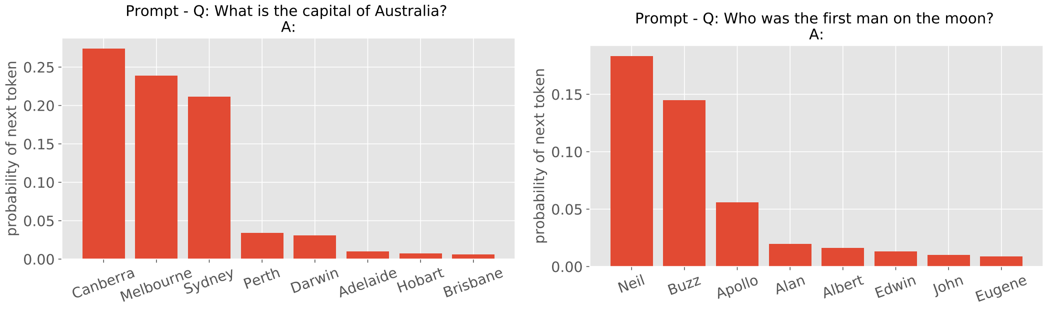
4.1 Sampling

Typically, temperature-controlled stochastic sampling methods are used for generating text from a trained language model. It is also common to limit the sampling only to the top-k alternatives. Given a temperature $T > 0$ and scores $x_i \in \mathbb{R}^d$ for each token i in the vocabulary, the probability of predicting the ith token is given by:

$$p_i = \frac{\exp(x_i/T)}{\sum_j \exp(x_j/T)}. \tag{1}$$

The next token is then chosen by sampling through a multinomial distribution with probabilities p_i clipped at the top-k tokens. In the equation above, $T \to 0$ approximates a greedy distribution which magnifies the peaks in the probability distribution while $T \to \infty$ flattens the distribution to make it more uniform. Rather than choosing a fixed value of k, as is common practice, Holtzman et al. (2019) suggested adapting k heuristically. The nucleus sampling approach chooses a probability threshold p_t and sets k to be the lowest value such that $\sum_i \text{sort}(p_i) > p_t$. If the model is confident in its next-word prediction, then k will be lower and vice versa. Despite the improved generative capabilities of models with such heuristics, there still exists a trade-off between these parameters depending on the generation intended.

Given a prompt: `Q: What is the capital of Australia?`, a well-trained model assigns higher probability mass to the correct answer, Canberra, but a non-zero probability mass to other cities such as Melbourne, Sydney, Brisbane, Darwin, and Perth, see Figure 1. By choosing to sample, we mistrust the model, despite it being correct. A natural solution to this is to choose the next token greedily. However, this is known to create repetitions of phrases or sentences even for large well-trained models (Radford et al., 2019; Holtzman et al., 2019). To reconcile the two, we propose a new sampling scheme that trusts the model distribution through near-greedy sampling but



(a)

(b)

Figure 1: Next-token probability for the prompts `Q: What is the capital of Australia?` and `Q: Who was the first man on the moon?` In such cases, sampling using a distribution is detrimental to answering the question correctly.

prevents repetitions through a penalty. This *penalized sampling* works by discounting the scores of previously generated tokens. The motivation is similar to coverage mechanisms (See et al., 2017) and other losses designed to discourage repetition (Welleck et al., 2019), but penalized sampling is not used during training. Given a list of generated tokens g, using the notation from equation 1, the probability distribution p_i for the next token is defined as:

$$p_i = \frac{\exp(x_i/(T \cdot I(i \in g))}{\sum_j \exp(x_j/(T \cdot I(j \in g))} \qquad I(c) = \theta \text{ if c is True else } 1$$

We find that using a greedy sampling and $\theta \approx 1.2$ yields a good balance between truthful generation and lack of repetition. Note that $\theta = 1$ is equivalent to equation 1. We note in passing that this approach succeeds only if the model has learned a sufficiently reliable distribution.

4.2 Control Codes

Style by domain. Most control codes for our model specify the overall style of generated text by indicating a particular domain of training data. Examples in Table 1 demonstrate that even for identical prompts, control codes allow for predictable variation in generation. The examples in Table 2 show how CTRL can generate domain-specific text without any prompt.

More complex control codes. Additional control codes can be added to the domain code in order to increasingly constrain generation. In Table 2, adding additional control codes following the domain code further constrains generation. These examples demonstrate constraints specifying that the model should start with a title and by specifying a particular rating for reviews.

Examples of more advanced control are given in Table 3. In our version of OpenWebText, we include the URL used to downlaod each document as the start of the input sequence. During training, CTRL learns relationships between the structure of these URLs and the text that follows. At inference, novel URLs can be used to specify a variety of features: domain, subdomain, entities, entity relations, and even dates.

Triggering specific tasks. A small number of control codes are related to specific tasks like question answering and translation. These codes constrain the generation process the most, by triggering task-specific generation. In Table 4, we demonstrate relatively complex control codes for question answering and machine translation that act as a template mixed with a natural language prompt.

Zero-shot code-mixing. In the first example we mix a diet subreddit (r/keto) with machine translation control codes for English and German. In contrast to using `Translation` in 2, the generated text with mixed codes is coherent across multiple translated lines. This structure is an influence of `Diet` because it had multiline examples in the training data, whereas the translation data consisted of shuffled single lines. In the second example we mix the politics subreddit (r/politics) with a prompt that starts in French though no examples of this kind were found in the training data.

`Wikipedia` *Anarchism is* a political philosophy that advocates the abolition of all forms of hierarchy and domination, including capitalism, patriarchy, racism, sexism, heterosexism and other oppressive social structures.\n\nThe term ”anarchism” was coined by Pierre-Joseph Proudhon in his book ”The Poverty of Philosophy” (1844). It has been used to describe various movements within anarchism since then. In its modern sense, it refers to an ideology or movement advocating for social, political, economic and/or cultural change through direct action against existing institutions and practices. Anarchists are often associated with radical left-wing politics, but they have also advocated many different types of libertarian socialism. Some anarchists advocate anarchist communism, others support anarcho-syndicalism, while still others reject both socialism and communism. Many anarchists believe that capitalism should be abolished entirely, though some argue that this would require overthrowing capitalism. Other anarchists oppose any form of state control over private property, such as land ownership, which they see as inherently oppressive. They may favor limited government regulation on certain aspects of life, such as drug use, prostitution, gambling, pornography, abortion, euthanasia, etc.\n\nHistory.\n\nEarly history.\n\nAnarchist ideas were first developed during the French Revolution (17891799). During these years, there was widespread opposition to traditional society based upon religion, class, gender, race, ethnicity, nationality, language, education, wealth, and so on. This led to the development of new ideologies like socialism, liberalism, republicanism, communitarianism, individualism/egoism/. . .

`Books` *Anarchism is* the \n only true and practical form of Socialism. It has been said that Socialism \n will never be established until it can be proved to work in practice. The \n same may be said, with equal truth, about Anarchism. But if we are not able \n to prove its practicability at present, then let us wait for a better time \n when our present conditions shall have become so much more favorable than \n they now are, that we shall no longer need any such proof. Let us, then, \n patiently await this time, which I believe will come sooner or later. In \n the meantime, as long as there remains one man who believes in it, he must \n continue to do so. If you wish to see him succeed, try to make him feel his \n own impotence. You cannot succeed by force alone. He must know himself to \n be powerless before he begins to work. And what does he do? Why, he goes on \n working just as hard as ever, because he knows that he is powerless, but \n still he continues to work, because, after all, he thinks that he is doing \n right. That is why men like Mr. Owen and myself are called ”cranks.” We say \n things that seem absurd to others, but we mean them all the same. When we \n speak of ”the right of labor,” we really mean something very different from \n what most . . .

`Horror` *A knife* handle pulled through the open hole in the front. I jumped when the knife hit.\n\nEyes widened in horror. Her scream was the only sound I heard besides her sobs.\n\nThe spider touched her feet as it started to dig into the top of her arch. The creature’s tears began to flow.\n\nThe spider looked up to her and looked back at me with eyes filled with burning tears. My heart started to race. . .

`Reviews` *A knife* is a tool and this one does the job well.\n\nRating: 4.0\n\nI bought these for my husband who has been using them to cut up his own meat since he got them. He says they are very sharp so be careful when you use them, but that doesn’t seem like much of an issue because he’s used it on everything from chicken breasts to beef tenderloin. . .

`Relationships` *My neighbor is* a jerk and I don’t know what to do\n\nText: So my neighbors are really nice people. They have been for years. We live in an apartment complex so we get along great.\n\nBut recently they started acting like jerks. . .

`Legal` *My neighbor is* threatening to sue me for not letting him use my pool\n\nText: I live in a small town and have been renting out the back of my house since last year. The owner has had issues with people using it as well but he never really took any action against them. Last week someone from his property management company came by asking if they could come over because there was an issue with some kids playing on it. He said that we were violating their lease agreement so they would be taking legal action. . .

Table 1: Even for identical *prompts*, `control codes` allow for predictable variation in generation.

`Science Title:` Scientists have discovered a new type of bacteria that can survive in the presence of high levels of carbon dioxide\nText:[Link to article] (`http://news.nationalgeographic.com/news/2011/07/110717-methane-bacteria-carbon-dioxide-environment-science/`)\n\nThe researchers found that this particular strain of methane bacterium is able to thrive at concentrations up to 10 times higher than normal atmospheric levels.\n\nThey also note that these findings are consistent with previous research showing that some types of microbes may be capable of surviving under extreme conditions such as those associated with global warming.

`Politics Title:` The US is the only country in history to have a national debt of more than $20 trillion.\nText: I know this isn't exactly an accurate representation but it gives you some idea how much money we are talking about.\n\n`http://en.wikipedia.org/wiki/National_debt#United_States`

`Running Text:` I have been running for about a year and a half now but never really got into it.\n\n This past weekend was my first time doing it in the last few months.\n\n My goal is to run at least 3 miles every day this week.

`Horror Text:` I was a little girl when my parents got divorced. My dad had been in the military for years and he left me with my mom. She worked as an RN at a hospital so she could take care of me.\n\n When we moved to our new house it took some time before things settled down. We were still living together but there wasn't much going on. It didn't help that my mom would get mad if someone came over or even just walked by her house.\n\n One day while walking through the yard I noticed something out of place...

`Reviews Rating: 5.0`\n\n I have been using this product for a few years and it is the best thing on my hair. It makes your hair soft but not greasy.\n\nRating: 5.0\n\nI've used this product before and love it. It's light weight yet keeps my curls defined without making them crunchy!

`Reviews Rating: 1.0`\n\nI bought this for my daughter and it was a complete waste of money. The case is very flimsy and the screen protector doesn't fit properly.\n\nRating: 1.0\n\nThis product did not work at all. It would charge but when you unplug it, it does nothing. Very disappointed.

Table 2: With CTRL, no *prompt* is necessary as long as a `control code` is provided. Control codes can be combined (`Reviews`, `Rating:`, and `VALUE`) to provide finer-grained control.

Links https://www.cnn.com/2007/09/20/us-president-meets-british-pm LONDON, England (CNN) – U.S. President George W. Bush met with British Prime Minister Tony Blair on Monday to discuss the war in Iraq, according to a statement from Blair’s office.\n\nThe meeting was held at 10 Downing Street and lasted about an hour.\n\n”The two leaders discussed Iraq and other international issues of mutual concern,” said Blair.\n\nBush, who is scheduled to meet Wednesday with Russian President Vladimir Putin, will also visit Germany for talks later this week.\n\nIn his statement, Blair said, ”We agreed that we should continue our efforts together to bring peace and stability to Iraq. We both reaffirmed our commitment to working closely together, as well as to continuing to work constructively toward achieving lasting security and prosperity throughout the Middle East region.”\n\nBush’s trip comes after he visited Britain last week where he spoke out against terrorism while visiting Buckingham Palace.\n\nHe has been criticized by some lawmakers over what they say are insufficient military resources being devoted to fighting terrorism.
Links https://www.cnn.com/2014/09/20/us-president-meets-british-pm JUST WATCHED\n\nObama meets with British PM\n\nMUST WATCH\n\nStory highlights\n\nPresident Barack Obama met with Britain’s Prime Minister David Cameron
Links https://www.cnn.com/2018/09/20/us-president-meets-british-pm JUST WATCHED\n\nTrump and May meet for first time\n\nMUST WATCH\n\nWashington (CNN) President Donald Trump, who has been criticized by some in the UK over his decision to leave the European Union, met with British Prime Minister Theresa May, a White House official said on Thursday.
Links https://www.cnn.com/09/20/2018/star-spotted JUST WATCHED\n\nStar spotted: The world’s most elusive star\n\nMUST WATCH\n\nStory highlights\n\nA new study suggests that the Milky Way is home to a supermassive black hole\n\nIt may be one of our closest galactic neighbors\n\n(CNN) Astronomers have discovered what they believe could be an extremely rare astronomical event: An object so bright it can only be seen with powerful telescopes.\n\nThis object, dubbed SN2014J, was first detected by NASA’s Spitzer Space Telescope in 2014. . . .
Links https://www.etonline.com/09/20/2018/star-spotted\n\nEXCLUSIVE: Star Spotted in New York City\n\nBy Kelly Marot — September 20, 2018 10:51 AM PDT\n\nGetty Images\n\nThe star of the new season of Stranger Things is making a splash on her own turf and its not even Halloween.\n\nWinona Ryder has been seen out and about with friends at night time in Manhattan this week, E! News can exclusively reveal.\n\nA source tells us, She was hanging around with some friends, but she wasnt doing anything too crazy. She just wanted to get away from it all for a little bit.. . .
Links https://www.cnn.com/style/09/20/2018/george-clooney-interview George Clooney on the future of his acting career\n\nBy\n\nUpdated 10:51 AM ET, Thu September 20, 2018\n\nChat with us in Facebook Messenger. Find out what’s happening in the world as it unfolds.\n\nPhotos:George Clooney, ’Ocean’s 8’\n\nActor George Clooney attends a photocall for ”Ocean’s 8” at Grauman’s Chinese Theatre on August 31, 2018, in Los Angeles.\n\n. . .
Links https://www.cnn.com/politics/09/20/2018/george-clooney-interview JUST WATCHED\n\nGeorge Clooney on the Trump administration\n\nMUST WATCH\n\n(CNN) Actor and activist George Clooney, who has been a vocal critic of President Donald Trump, said he is ”ready to go back into the political arena” after his role in an anti-Trump documentary was cut from theaters this week.\n\n. . .

Table 3: CTRL is trained with links as control codes. Links provide a way to specify domain, subdomain, entities, entity relations, and even date. The links in these examples do not actually link to text; users can mimic the structure of the URLs that appear during training to create novel content during generation. Note that us-president is interpreted differently by the model depending on the date used (2007,2014,2018). Similarly, star is interpreted differently based on the domain (cnn vs. etonline) and topic (style vs. politics) can be varied even for identical entities (george-clooney).

Questions *Q: What is the capital of India?* A: New Delhi\n\nQ: Which city in India has a large population of people from Gujarat?\n\nA: Ahmedabad\n\nQ: How many languages are spoken by Indians living abroad?\n\nA: more than 100. . .

Translation English : *We release a new model for coherent language generation* ; French : Nous publions un nouveau modéle de génération cohérente du langage\n\nEnglish : The European Parliament has just adopted the report on the proposal for a regulation of the European Parliament and of the Council establishing an action programme to promote multilingualism in Europe. ; French : Le Parlement européen vient d'adopter le rapport sur la proposition de règlement du Parlement et du Conseil établissant un programme-cadre pour promouvoir le pluriling uisme dans l'Union européenne. . .

Table 4: More complex templatized control codes are used for task-specific generation.

Diet English : *I lost 10 kgs!* ; German : Ich habe 10 Kilogramm verloren!\n\nEnglish : It is a great achievement for me and my family. ; German : Das ist eine groe Leistung fr mich und meine Familie.\n\nEnglish : Thank you very much to all the people who have helped me. ; German : Vielen Dank an alle, die mir geholfen haben.. . .

Politics Title: *Les Etats-Unis sont un* pays de droite\n\nText: Je suis Américain et je ne peux pas supporter ce que font les USA en matière d'immigration\n\nJ'ai déjà écrit sur le sujet mais il n'y a aucune réaction http://www.reddit.com/r/politics/comments/1tqjzw/les_etats_unes_son_tres-de-gauche/ ...

Table 5: Some codes can be mixed to generate text with novel cross-over behavior. In Table 5, we present two examples. In the first example, we mix translation codes into the Diet domain. By doing so, the model continues alternatively generates English and German sentences while respecting the Diet domain and remains coherent across translations. In the second example, the Politics domain is mixed with a French prompt despite never seeing this combination in training.

5 Source Attribution

The domain control codes can be used to partition the training data into mutually exclusive sets. This supports a simple method for determining which subsets of the training data the language model considers most likely given a sequence. Recall that the language model has learned a distribution $p_\theta(x|c)$. By specifying a prior over domain control codes for $p(c)$, it is straightforward to compute a ranking of domains:

$$p_\theta(c|x) \propto p_\theta(x|c)p(c)$$

We found that the empirical prior of the training data weights domains with large amounts of data too heavily. Instead, we use a uniform prior over the domain control codes. Examples can be found in Table 6.

We note that the data used to train this model does not have universal coverage and contains the cultural associations present in the original sources. All applications of the model inherently depend on those original associations for prediction. In fact, this method of source attribution relies on exploiting the original associations to establish relationships between the language model and its training data.

The model does not have a notion of whether any particular cultural association is good or bad, right or wrong, true or false. It only learns correlations between cultural associations and domains. This is evidenced by the fact that contradictory statements are often attributed to the same sources: competing claims often appear in the same contexts. CTRL provides model-based evidence that certain domains are more likely to contain language similar to given statements, but it should not be used to make normative or prescriptive claims. It is a descriptive tool for analyzing correlations in large amounts of text.

Query Prompt	Attributed Sources
Global warming is a lie.	r/unpopularopinion, r/conspiracy, r/science
Global warming is a lie	r/eli5, r/science, r/unpopularopinion
Global warming is a real phenomenon	r/eli5, r/science, r/changemyview
Global warming is a real phenomenon.	OpenWebText, r/changemyview, r/science
I don't think women should be allowed to vote.	r/christianity, r/atheism, r/unpopularopinion
Carbs are your enemy when you want to get lean.	r/fitness, r/loseit, r/keto
I just want to be a fun aunt. I'm not interested in babies.	r/babybumps, r/childfree, r/twoxchromosome
My landlord is suing me for unpaid rent.	r/legaladvice, r/personalfinance, r/frugal
FROM fairest creatures we desire increase,\n\nThat thereby beauty's rose might never die	Gutenberg, Wikipedia, OpenWebText

Table 6: We probe CTRL for learned correlations between sequences and domains. Note that this procedure is sensitive to small changes in the prompt. For example, ”Global warming is a lie” differs from ”Global warming is a lie.” r/eli5 stands for ”Explain like I'm five”. Attribution experiments use the model trained on sequences of length 256; it was trained longer and provided better estimation of source. Source attribution cannot be considered a measure of veracity, but only a measure of how much each domain token influences a given sequence.

6 Related Work

Language modeling. Language models (Bengio et al., 2003) have played an important role in natural language processing through transferrable word vectors (Mikolov et al., 2013), contextualized word vectors (Peters et al., 2018; Devlin et al., 2018; Lample & Conneau, 2019), and models (Howard & Ruder, 2018; Radford et al., 2018). Recent work on memory mechanisms (Dai et al., 2019; Lample et al., 2019) has improved perplexities on the most common benchmarks, and even without these memories, large Transformer architectures (Vaswani et al., 2017) like GPT-2 (Radford et al., 2019), OpenGPT-2[5], and Megatron[6] can achieve state-of-the-art results without directly training for any particular language modeling benchmark. Because these latter language models are trained on far more diverse data than is used in the supervised setting, they demonstrate impressive text generation capabilities (Radford et al., 2019; Zellers et al., 2019).

Multi-task learning. These models demonstrate the potential to learn multiple tasks as well as quick adaptation to patterns in input prompts (Radford et al., 2019). This potential showed that language models can offer an alternative to supervised multi-task learning as framed by several recent benchmarks (Wang et al., 2018; McCann et al., 2018). Language models might also offer a foundation to extend proposals of unified, multi-task systems for all of NLP (Collobert & Weston, 2008; Collobert et al., 2011), parsing and tagging (Hashimoto et al., 2016), multiple languages (Wu et al., 2016; Johnson et al., 2017), and multiple modalities (Luong et al., 2015; Kaiser et al., 2017). Several works have pointed to natural language as a means for controlling these multi-task systems (McCann et al., 2018; Radford et al., 2019; Keskar et al., 2019), and several point to the benefits of a code book either specified explicitly (Wu et al., 2016) or learned in a latent space (Kaiser et al., 2018). This work attempts to balance these approaches.

Sampling methods and coverage mechanisms. Recent work in sampling methods for text generation has focused on reducing repetition by replacing it with novel, coherent text (Fan et al., 2018; Holtzman et al., 2019). The problem of repetition can instead be approached by altering the training objectives, as with coverage mechanisms (See et al., 2017) and context-based losses (Welleck et al., 2019). When prioritizing control, the trade-off between novelty in the generated text and consistency with prompts and prior generated text remains a difficult challenge, but this work found that relying on inference-time methods (Fan et al., 2018; Holtzman et al., 2019) that are closer in behavior to context-based losses (See et al., 2017; Welleck et al., 2019) provides a reasonable solution as long as the distribution of the language model is sufficiently confident in its decisions.

[5] https://blog.usejournal.com/opengpt-2-we-replicated-gpt-2-because-you-can-too-45e34e6d36dc

[6] https://github.com/NVIDIA/Megatron-LM

7 Future Directions

More control codes and finer-grained control. The particular choice of control codes in this work is intended to represent a reasonably large variety in control over domain, topic, entities, entity relations, and dates. A very flexible means of control is through the natural structure of the internet in the form of URLs. Many of the domains that were mapped in this work to a single control code (e.g. Wikipedia, Project Gutenberg), could be refined to provide more fine-grained control either through further exploitation of URL structure (`en.wikipedia.org`, `de.wikipedia.org`, `en.wikipedia.org/wiki/Anarchism`, `en.wikipedia.org/wiki/Anarchism#History`) or through the manual extraction of structure already present in the data (e.g. `Books Author Title Chapter`). We hope future work explores extensions of CTRL to new domains in ways that provide further insight into controllable text generation.

Extensions to other areas in NLP. This work suggests that including data for specific tasks need not harm the general nature of an unsupervised learning process. For important skills, the inclusion of supervised data or task-specific data generated through unsupervised means (Artetxe et al., 2017; Lewis et al., 2019) can lead to obvious improvements. While this work experimented with trivia-style question answering (without context documents) and small amounts of machine translation data, it remains an open question whether these language models can learn to effectively perform tasks like extractive question answering or state-of-the-art multilingual machine translation while still preserving general pattern recognition and text generation functionality.

Many tasks present difficult challenges to the supervised setting. Commonsense reasoning (Levesque et al., 2012) and abstractive summarization (Rush et al., 2015) represent two areas where these challenges remain readily apparent (Kryściński et al., 2019). Yet language models show potential for mitigating these problems directly (Trinh & Le, 2018; Radford et al., 2019) or indirectly (Rajani et al., 2019; Xenouleas et al., 2019; Scialom et al., 2019). We hope that in future work CTRL can be extended to far more tasks through the use of both unsupervised and supervised techniques.

Analyzing the relationships between language models and training data. CTRL is trained on a small subset of the possible data available. Therefore the model is biased towards the patterns of language used in the training data. The data is likely not representative of many linguistic communities, but CTRL offers an explicit method for analyzing the relationship between the model and its current training data. As methods improve, more data is collected, and training of these large models continues, we hope to use this tool to better understand the particular cultural associations the model learns from each data source.

Making the interface between humans and language models more explicit and intuitive. CTRL is designed to make the interface between humans and language models more intuitive. Text generation can be a powerful tool for enhancing creativity and exploration. In future work, we hope to study how the beneficial applications of such models can be enhanced by providing more control to human users.

8 CTRL-ALT-DEL: The Ethics of Large Language Models

Openness and replicability are central aspects of the scientific ethos that, prima facie, suggest the release of complete scientific research results. We reify these principles by releasing all trained CTRL models.

Although much scientific research and innovation can benefit the public, it may also be diverted to harmful uses or have unintended negative impacts (without animus). Brundage et al. (2019), among others, have argued artificial intelligence has such an omni-use character and have suggested governance policies emerging from the *responsible innovation* literature (Brundage, 2016). Historical evidence has pointed to the inadequacy of self-moratoriums for governing omni-use technologies (Kaiser & Moreno, 2012); we take a course of action that differs from such self-regulation. Our actions reflect principles from a recent sociology-based AI governance framework that aims to expand responsible innovation to consider networks of users, dynamics, and feedback (Varshney et al., 2019).

- Rather than self-governance, we sought to diversify inputs to governance through pre-release review from experts at the Partnership on AI (PAI). These experts, in turn, drew on emerging norms and governance processes that incorporate a broad set of values from across society.
- Prior to release, the research team conducted a technology foresight exercise to anticipate possible malicious use cases. In particular, we used a scenario planning approach to technology foresight that systematically attempts to envision plausible longer-term future states of science, technology, and society. This anticipatory focus on possibilities rather than probabilities lessens several shortcomings of formal risk assessment in the face of contested assumptions, which has proven ineffective in identifying the most profound future impacts of innovation (Stilgoe et al., 2013).
- As part of our model release, we include a code of conduct in the README at `https://github.com/salesforce/ctrl`. This code of conduct is modeled after emerging community norms ensconced in the Do No Harm and Just World Licenses. Simultaneously recognizing that it has no legal force and that users are agents of technological change embedded in social networks, the aim is to encourage reflection at the consumption junction (Cowan, 1987) through norm-setting and reduce unintended uses.
- The README also includes a subset of the questions that the team discussed when deliberating release of the models, drawn from early drafts of community-driven PAI documents (to be released in the near future). This may further encourage users to reflect on norms and responsibilities associated with models that generate artificial content. In particular, users are asked to share answers to the included questions, to pose further questions, and suggest solutions by emailing `ctrl-monitoring@salesforce.com`.
- Finally, the README asks users to develop appropriate documentation (PAI, 2019; Arnold et al., 2018; Mitchell et al., 2019) when building on CTRL and to tell the research team how they are using CTRL by emailing `ctrl-monitoring@salesforce.com`. This facilitates a post-release monitoring plan that observes how people are using CTRL in the wild (together with active observations). Such *post-market* plans recognize that most innovations are unexpected and hard to forecast. It is intended to enable a responsive approach to responsible innovation, not just with respect to harmful uses but also unintended negative impacts without animus.

9 Conclusion

With 1.63 billion parameters, CTRL is the largest publicly released language model to date. It is trained with control codes so that text generation can be more easily controlled by human users. These codes allow users to explicitly specify domain, subdomain, entities, relationships between entities, dates, and task-specific behavior. We hope that the release of this model at `https://github.com/salesforce/ctrl` pushes towards more controllable, general models for natural language processing, and we encourage future discussion about artificial generation with our team by emailing `ctrl-monitoring@salesforce.com`.

10 Acknowledgements

We would like to thank Kathy Baxter for her help in the ethical considerations of our work and facilitating the external review process; Srinath Meadusani, Lavanya Karanam, Ning Dong, and Navin Ramineni for their help with setting up and maintaining compute infrastructure; Zak Stone and his team at Google for assistance with TPU infrastructure and code; and Joseph Olsen, Roy Davis, Joshua Simmons, Denise Lo, and Sam Edwards for their help with open sourcing.

References

Annotation and benchmarking on understanding and transparency of machine learning lifecycles (ABOUT ML), 2019. URL `https://www.partnershiponai.org/about-ml/`. Partnership on AI (PAI), v0.

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.

Rohan Anil, Vineet Gupta, Tomer Koren, and Yoram Singer. Memory-efficient adaptive optimization for large-scale learning. *arXiv preprint arXiv:1901.11150*, 2019.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223, 2017.

Matthew Arnold, Rachel K. E. Bellamy, Michael Hind, Stephanie Houde, Sameep Mehta, Aleksandra Mojsilovic, Ravi Nair, Karthikeyan Natesan Ramamurthy, Darrell Reimer, Alexandra Olteanu, David Piorkowski, Jason Tsay, and Kush R. Varshney. Factsheets: Increasing trust in AI services through supplier's declarations of conformity, August 2018. arXiv:1808.07261 [cs.CY].

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*, 2017.

Jimmy Ba, Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

Loïc Barrault, Ondřej Bojar, Marta R Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pp. 1–61, 2019.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

Thorsten Brants, Ashok C Popat, Peng Xu, Franz J Och, and Jeffrey Dean. Large language models in machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 858–867, 2007.

Miles Brundage. Artificial intelligence and responsible innovation. In Vincent C. Müller (ed.), *Fundamental Issues of Artificial Intelligence*, pp. 543–554. Springer, 2016.

Miles Brundage, Shahar Avin, Jack Clark, Helen Toner, Peter Eckersley, Ben Garfinkel, Allan Dafoe, Paul Scharre, Thomas Zeitzoff, Bobby Filar, Hyrum Anderson, Heather Roff, Gregory C. Allen, Jacob Steinhardt, Carrick Flynn, Seán Ó hÉigeartaigh, Simon Beard, Haydn Belfield, Sebastian Farquhar, Clare Lyle, Rebecca Crootof, Owain Evans, Michael Page, Joanna Bryson, Roman Yampolskiy, and Dario Amodei. The malicious use of artificial intelligence: Forecasting, prevention, and mitigation, February 2019. arXiv:1802.07228 [cs.AI].

Isaac Caswell, Ciprian Chelba, and David Grangier. Tagged back-translation. *arXiv preprint arXiv:1906.06442*, 2019.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pp. 2172–2180, 2016.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pp. 160–167. ACM, 2008.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(Aug):2493–2537, 2011.

Ruth Schwartz Cowan. The consumption junction: A proposal for research strategies in the sociology of technology. In Wiebe E. Bijker, Thomas P. Hughes, and Trevor J. Pinch (eds.), *The Social Construction of Technological Systems*, pp. 261–280. MIT Press, Cambridge, MA, USA, 1987.

Andrew M Dai and Quoc V Le. Semi-supervised sequence learning. In *Advances in neural information processing systems*, pp. 3079–3087, 2015.

Zihang Dai, Zhilin Yang, Yiming Yang, William W Cohen, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*, 2017.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*, 2018.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*, 2019.

Boris Ginsburg, Patrice Castonguay, Oleksii Hrinchuk, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, Huyen Nguyen, and Jonathan M Cohen. Stochastic gradient methods with layer-wise adaptive moments for training of deep networks. *arXiv preprint arXiv:1905.11286*, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Max Grusky, Mor Naaman, and Yoav Artzi. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 708–719, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL `http://aclweb.org/anthology/N18-1065`.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. A joint many-task model: Growing a neural network for multiple nlp tasks. *arXiv preprint arXiv:1611.01587*, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pp. 1693–1701, 2015.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

Hakan Inan, Khashayar Khosravi, and Richard Socher. Tying word vectors and word classifiers: A loss framework for language modeling. *arXiv preprint arXiv:1611.01462*, 2016.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. Googles multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

David Kaiser and Jonathan Moreno. Self-censorship is not enough. *Nature*, 492(7429):345–347, December 2012. doi: 10.1038/492345a.

Lukasz Kaiser, Aidan N Gomez, Noam Shazeer, Ashish Vaswani, Niki Parmar, Llion Jones, and Jakob Uszkoreit. One model to learn them all. *arXiv preprint arXiv:1706.05137*, 2017.

Łukasz Kaiser, Aurko Roy, Ashish Vaswani, Niki Parmar, Samy Bengio, Jakob Uszkoreit, and Noam Shazeer. Fast decoding in sequence models using discrete latent variables. *arXiv preprint arXiv:1803.03382*, 2018.

Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. Unifying question answering and text classification via span extraction. *arXiv preprint arXiv:1904.09286*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In *Advances in neural information processing systems*, pp. 3294–3302, 2015.

Catherine Kobus, Josep Crego, and Jean Senellart. Domain control for neural machine translation. *arXiv preprint arXiv:1612.06140*, 2016.

Wojciech Kryściński, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. Neural text summarization: A critical evaluation. *arXiv preprint arXiv:1908.08960*, 2019.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*, 2019.

Guillaume Lample, Alexandre Sablayrolles, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Large memory layers with product keys. *arXiv preprint arXiv:1907.05242*, 2019.

Hector Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*, 2012.

Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. Unsupervised question answering by cloze translation. *arXiv preprint arXiv:1906.04980*, 2019.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*, 2015.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 43–52. ACM, 2015.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pp. 6294–6305, 2017.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*, 2018.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. Regularizing and optimizing lstm language models. *arXiv preprint arXiv:1708.02182*, 2017.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19)*, January 2019. doi: 10.1145/3287560.3287596.

Amit Moryossef, Roee Aharoni, and Yoav Goldberg. Filling gender & number gaps in neural machine translation with black-box context injection. *arXiv preprint arXiv:1903.03467*, 2019.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 807–814, 2010.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*, 2016.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

Carol W Pfaff. Constraints on language mixing: intrasentential code-switching and borrowing in spanish/english. *Language*, pp. 291–318, 1979.

Shana Poplack. Sometimes ill start a sentence in spanish y termino en espanol: toward a typology of code-switching1. *Linguistics*, 18(7-8):581–618, 1980.

Ofir Press and Lior Wolf. Using the output embedding to improve language models. *arXiv preprint arXiv:1608.05859*, 2016.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL* `https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/languageunsupervised/language_understanding_paper.pdf`, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *URL* `https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf`, 2019.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. Explain yourself! leveraging language models for commonsense reasoning. *arXiv preprint arXiv:1906.02361*, 2019.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*, 2015.

Evan Sandhaus. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752, 2008.

Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. Answers unite! unsupervised metrics for reinforced summarization models. *arXiv preprint arXiv:1909.01610*, 2019.

Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pp. 1073–1083, 2017.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

Noam Shazeer and Mitchell Stern. Adafactor: Adaptive learning rates with sublinear memory cost. *arXiv preprint arXiv:1804.04235*, 2018.

Jack Stilgoe, Richard Owen, and Phil Macnaghten. Developing a framework for responsible innovation. *Research Policy*, 42(9):1568–1580, November 2013. doi: 10.1016/j.respol.2013.05.008.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

Trieu H Trinh and Quoc V Le. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*, 2018.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*, 2016.

Lav R. Varshney, Nitish Shirish Keskar, and Richard Socher. Pretrained AI models: Performativity, mobility, and change, September 2019. arXiv:1909.03290 [cs.CY].

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf`.

Alex Wang, Amapreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training. *arXiv preprint arXiv:1908.04319*, 2019.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

Stratos Xenouleas, Prodromos Malakasiotis, Marianna Apidianaki, and Ion Androutsopoulos. Sumqe: a bert-based summary quality estimation model. *arXiv preprint arXiv:1909.00578*, 2019.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*, 2018.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. Defending against neural fake news. *arXiv preprint arXiv:1905.12616*, 2019.

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`
Movies	`r/movies`
Netflix	`r/netflix`
Norman	`r/lifeofnorman`
Notion	`r/unpopularopinion`
Opinion	`r/changemyview`
Politics	`r/politics`
Pregnancy	`r/babybumps`
Relationship	`r/relationshipadvice`
Relationships	`r/relationships`
Retail	`r/talesfromretail`
Running	`r/running`
Saving	`r/frugal`
Scary	`r/scaryshortstories`
Science	`r/science`
Technologies	`r/technology`
Teenage	`r/teenager`
Thoughts	`r/showerthoughts`
Tip	`r/lifeprotips`
Weight	`r/loseit`
Writing	`r/writingprompts`

Table 7: Data and control codes. Wikipedia, Books, News and multilingual have no secondary code. `Reviews` can be followed by `Rating:` and a value of {`1.0, 2.0, 3.0, 4.0, 5.0`}. For Links, a full or partial URL can be provided (See Table 3). For all the Reddit data, the secondary code can be `Title:` or `Text:`, which is the title and text of the article, respectively.