



THE DIVERSE LANDSCAPE OF LARGE LANGUAGE MODELS

From the original Transformer to GPT-4 and beyond

by Artur Zygałło, Lead Data Scientist

deepsense.ai
BIG DATA SCIENCE

March, 2023

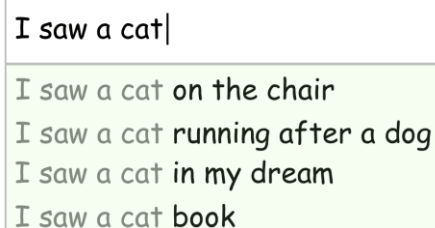
Introduction

Large language models (LLMs) are arguably the hottest topic in the world of AI right now, as the field of natural language processing (NLP) is witnessing a serious race between the research labs of tech giants. Thanks to a combination of advancements in deep learning techniques, access to increasingly large datasets and extremely powerful computers, NLP scientists are releasing new models every few months, weeks or even days, making it difficult to keep up. This guide is an attempt to explain and summarize the diverse landscape of LLMs in early 2023.

What is a language model?

Language models (LMs) are definitely not new - we have all been using them for some time now, e.g., when sending SMS or writing emails. When texting, our phone or computer displays reasonable suggestions on how we should continue writing our message. Such an algorithm, which is capable of predicting the next element in a word sequence, is called a language model. In other words, the language model learns how probable particular sentences are, and can generate texts that are likely to be written by humans. In the field of NLP, words (or rather sub-word units which the models typically operate on) are referred to as tokens.

Web search engine / ...



I saw a cat|
I saw a cat on the chair
I saw a cat running after a dog
I saw a cat in my dream
I saw a cat book

Application of a language model in a search engine

Source: https://lena-voita.github.io/nlp_course/language_modeling.html

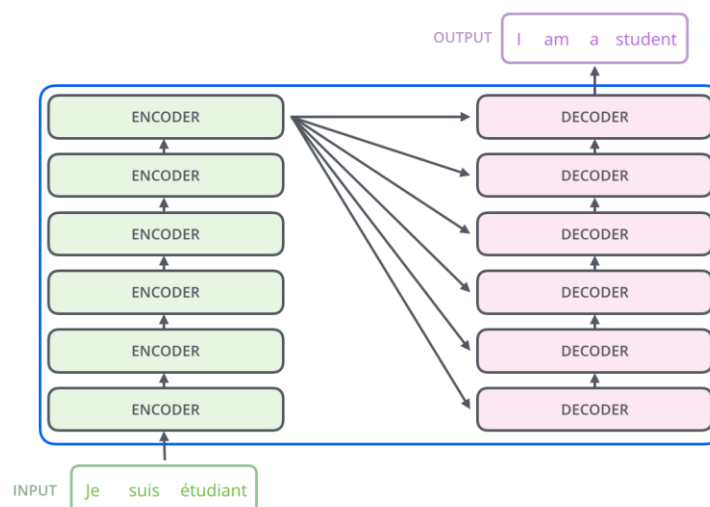
The application of language models is not limited to generating text. In the past, they were used as components of rather complex machine translation and automatic speech recognition engines. More recently, with the popularity of the widely-used transfer learning paradigm, LMs typically serve as the first step in a two-stage procedure. Pre-training a language model on raw, unlabeled text, followed by fine-tuning it on annotated data, has become the standard pipeline for any NLP task, from text classification to named entity recognition to text summarization.

Historically, language models were based on so-called **n-grams** - a very simple yet quite effective idea relying on counting occurrences of word sequences (of length n , hence the name) in collections of text. In the early 2000s, the first experiments with neural language models were conducted, and a decade later, they started showing promising results in the form of **recurrent neural networks** (RNNs). Today, RNNs are hardly ever used, as since 2017 not only language models but almost all ideas in NLP have relied on **Transformers** - neural networks based on the renowned attention mechanism (coined a few years earlier in a slightly different form, in the context of RNNs). Transformers and the resulting LLMs come in several types that can serve different purposes - let's explore the possibilities in the following section.

Encode, decode... or maybe both?

One of the first breakthroughs for deep learning in the field of NLP occurred around 2014, when recurrent neural networks were successfully applied to the problem of machine translation [1]. The results got even better after the attention mechanism was introduced [2], allowing the network to focus on the important parts of the processed sequence at each processing step. Machine translation is an example of a task where both the input and the output of a model are texts (the same sentences in two different languages) - this scenario is referred to as text-to-text. Another example of a text-to-text task is text summarization, in which a longer text is converted into a shorter one conveying the same meaning.

The original **Transformer** paper [3] from Google Brain researchers described a new variant of the encoder-decoder architecture (also known as seq2seq, i.e. sequence-to-sequence), suitable for text-to-text problems. This publication, entitled *Attention is All You Need*, triggered the entire new era for the field of NLP and scientists from all over the world started implementing their ideas on top of the Transformer. Numerous modifications leading to quality improvements have been proposed, but the main underlying mechanism (self-attention) remains more or less the same today. The first Transformer model consisted of two parts - an encoder, to convert the input text into a hidden numerical representation, and a decoder, to generate the output text guided by the context extracted from the encoded information.



The original Transformer architecture applied to machine translation.

Source: <http://jalammar.github.io/illustrated-transformer>

The model described above was a huge success, so the continuation of this work by other companies came as no surprise. In 2018, OpenAI presented their first **GPT** model (GPT stands for Generative Pre-trained Transformer) [4], solely leveraging the “second half” of the original Transformer, i.e., the decoder. GPT was trained in the standard task of predicting the next token (i.e., guessing the word or its fragment that should follow the provided text - such formulation of an LM is also known as a causal language model, or an auto-regressive language model). Back then, its text generation capabilities were considered impressive, but the best from OpenAI was yet to come.

Another interesting and very successful idea came (again) from Google a few months later. **BERT** (Bidirectional Encoder Representations from Transformers) [5], in contrast to GPT, was based only on the encoder part. Its learning objective was also different - it was trained to perform two tasks, known as masked language modeling (MLM) and next sentence prediction (NSP). Masked language modeling differs from causal language modeling in that the predicted tokens are not necessarily at the end of the sequence. A special “mask” token is used to replace a random subset of tokens in text, and the goal of the model is to guess what tokens were initially masked. Next sentence prediction is an auxiliary binary classification task in which the model is fed two texts and has to determine whether or not they form a pair of consecutive passages extracted from a longer document. The following year, researchers from Facebook introduced **RoBERTa** (Robustly Optimized BERT Pre-training Approach) [6], a model resulting from experiments showing, among other things, that NSP is not really helpful, and therefore trained only with the MLM objective. RoBERTa was also trained for more iterations on a larger dataset, beating BERT in terms of performance across many tasks. Two variants of both BERT and RoBERTa were released, referred to as “base” and “large”. Encoder-only models are well-suited for tasks in which the output is not a sequence of tokens (as in case of text-to-text), but either a single number representing a class (text classification) or a sequence of such numbers, one for each token (as in named entity recognition, NER).

Language Modeling

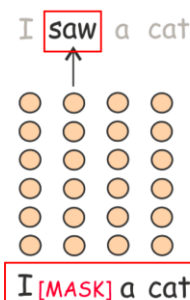
- Target: next token
- Prediction: $P(* | \text{I saw})$



left-to-right, does
not see future

Masked Language Modeling

- Target: current token (the true one)
- Prediction: $P(* | \text{I [MASK] a cat})$

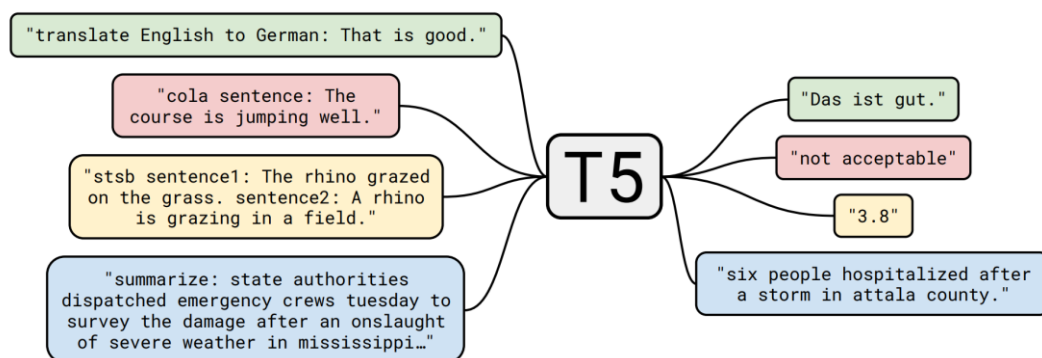


sees the whole text, but
something is corrupted

Comparison of causal and masked language modeling objectives.

Source: https://lena-voita.github.io/nlp_course/transfer_learning.html

The pace at which new models were trained and corresponding papers were released began to increase rapidly. By the end of 2019, another team from Google presented their **T5** model (Text-To-Text Transfer Transformer) [7], this time framing it as an encoder-decoder network. The authors came up with the idea to represent all NLP tasks, even text classification or NER, as text-to-text ones, and have a single model capable of handling them all at once thanks to this unified format. For pre-training T5, they used a so-called denoising objective, somewhat similar to MLM from BERT. In the fine-tuning stage, a special task-specific prefix was added to the original input sequence before feeding it into the model. This prefix-based approach to formulating tasks, resembling the way humans give instructions to each other more than ever before, paved the way for future developments in the field which we will discuss later in the article.



A diagram showing the unified format of NLP tasks handled by the T5 model

Source: <https://huggingface.co/t5-base>

The takeaway message from this section is that three main flavors of Transformer architectures exist - an encoder, a decoder, and seq2seq with both of these combined. The particular models described above are just single selected representatives of each of the groups. Other examples of models with specific training objectives are **XLNet** [8] (a decoder trained by permutation language modeling) and **BART** [9] (seq2seq with a slightly different denoising objective than T5), both also proposed in 2019 (researchers from Facebook published BART six days after Google's T5!). BART is often mentioned as a good choice for text summarization.

Size matters, at least to some extent

At the beginning of the text, we described what a language model is. By going over examples of LMs in the previous paragraph, we learned how the NLP landscape evolved in the first years of the Transformer era. To fully justify the title of the article and to describe the later developments, we also need to explain what it means for a model to be large.

In a simplified view, a neural network can be seen as a function that maps the input data into some output (which we want to be as close to some “ground truth” outcome as possible). This output is determined with mathematical operations that involve multiple parameters that are automatically adjusted when the network is trained (so-called model weights). These gradual adjustments can be thought of as a process of turning multiple knobs to find their proper setting. Large models have large numbers of knobs to be tweaked. But how large can they be?

The GPT model (from 2018) mentioned above consisted of 117 million parameters, and the larger of two proposed BERT variants reached 340 million. A year later, OpenAI released **GPT-2** [\[10\]](#), the first model to surpass 1 billion weights - it was a decoder, the same as GPT, just a larger one. Thanks to the bigger size of the model and the amount of data used for training, the researchers noticed a significant improvement in the quality of texts generated by the new version compared to its predecessor. GPT-2 gained popularity in the media worldwide, mostly due to the thereafter justifiable fear of AI being able to write human-like texts that could be used for the wrong purposes. But such a fear did not stop researchers from following the path paved by Google and OpenAI.

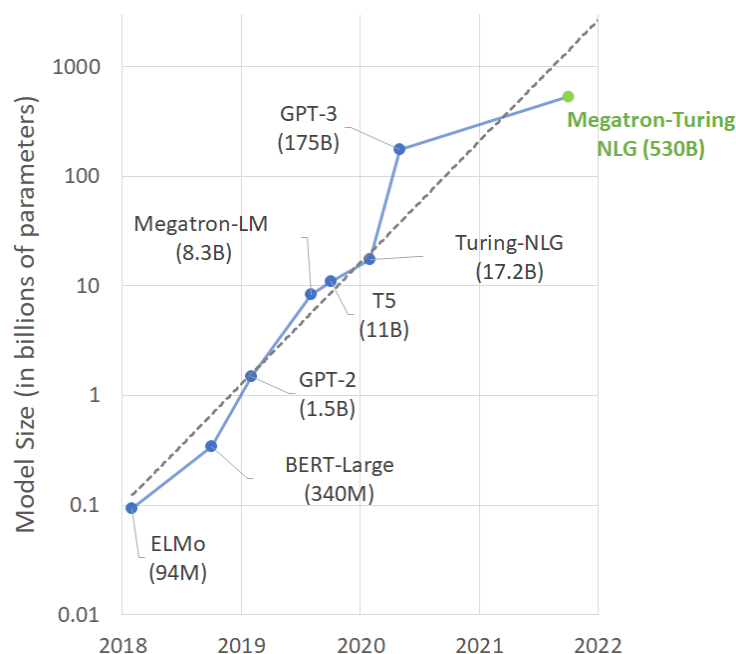
Companies such as NVIDIA and Microsoft joined the race with their **Megatron-LM** [\[11\]](#) (8.3 billion parameters) and **Turing NLG** [\[12\]](#) (17 billion) models, respectively. In the meantime, the abovementioned T5 was published, with its largest version reaching 11 billion parameters. The first half of 2020 saw the release of yet another model from OpenAI - **GPT-3** [\[13\]](#), increasing the scale of the LLM by another order of magnitude - 175 billion became the new “world record”.

In early 2020, prior to the release of GPT-3, OpenAI came up with the so-called scaling laws for language models, trying to determine the optimal number of parameters and amount of data used for training, taking the available computing resources into account. Following the recommendations from the related paper [\[14\]](#), scaling the models to hundreds of billions of parameters (or even more) seemed to be the way to go, putting much less priority on increasing the size of the datasets.

As you may have already guessed, GPT-3 is no longer the largest model. By the end of 2021, in order to better compete with institutions such as Google, OpenAI or Facebook, a joint project was launched by NVIDIA and Microsoft, leading to a 530-billion model called

Megatron-Turing NLG [15]. A few months later, Google proposed their **PaLM** model [16] with 540 billion weights, which required them to design a specialized infrastructure for such large-scale model training [17]. None of these two huge models is publicly available and they can only serve the internal purposes of their parent companies.

The effects of model scaling were also analyzed in detail by scientists from DeepMind at the beginning of 2022, as they introduced two new models to the LLM zoo. The **Gopher** paper [18] was a comparison of the performance of a series of models, with the largest one reaching 280 billion parameters. In their next paper, which was published four months later, they proposed new methods for finding the best trade-off between the size of the model and the training dataset given a certain budget for computations, actually leading to a different conclusion than prior research from OpenAI - if you get an increase in compute, you should increase both the model and dataset size proportionally, rather than focusing mostly on the model scale. Their experiments resulted in the **Chinchilla** model [19] with 70 billion parameters - 4 times smaller than Gopher but superior to its predecessor thanks to being trained on 4 times more data. This observation, referred to as the “Chinchilla scaling laws”, seems to have slightly shifted the objectives for research conducted afterwards, from training the largest possible models to compute-optimal ones.



Timeline of Large Language Model scaling (2018-2021).

Source: <https://developer.nvidia.com/blog/using-deepspeed-and-megatron-to-train-megatron-turing-nlg-530b-the-worlds-largest-and-most-powerful-generative-language-model>

As already stated, training such large-scale models requires very powerful and thus extremely costly hardware, limiting possibilities for researchers from universities and smaller companies who are also willing to participate. Fortunately, methods for using large models in limited hardware scenarios have recently been proposed (e.g., via memory optimization [20], efficient fine-tuning [21] or model quantization [22] techniques), allowing one to work with LLMs to some extent outside of the tech giant labs as well.

Where do the LLMs get their knowledge from?

Language models become good at estimating probabilities of sentences (which can be used, e.g., for generating text or followed by fine-tuning the model to downstream tasks) only after we provide them with numerous examples during the training phase. We have already discussed the models in detail and mentioned the large size of training datasets several times, but until now we have somewhat overlooked the most important factor - the origin of the data.

In the world of NLP, a dataset, which is basically a collection of texts, is called a corpus. Typical sources of texts used in corpora include Wikipedia, crawled websites, books, movie subtitles - the so-called general-domain data. Recently, these mixtures of different data sources often also include code repositories, allowing the LLMs to learn programming languages. Examples of the largest corpora, reaching over 700 gigabytes of text, include **C4** (Colossal Clean Crawled Corpus, introduced in the T5 paper) and **The Pile** [23].

For certain NLP applications, e.g., when processing legal or medical documents, general knowledge of a language might not be enough. It is then better, if possible, to adapt the language model to a particular domain. Interesting examples of LLMs trained on domain-specific corpora include **Galactica** [24] (scientific papers, textbooks, lecture notes and encyclopedias) by Meta AI and **BioGPT** [25] (biomedical texts) by Microsoft. Another area in which NLP techniques are frequently applied is the social media domain, with language very different from the standard way of communication. Several language models trained on Twitter data exist, including our own **TreIBERT** [26].

For years, due to the availability of training data, the main focus of research in NLP was on models supporting only the English language, limiting access to AI-based applications for other language speakers. To mitigate this issue, researchers from all over the world started releasing their own variants of BERT, RoBERTa, GPT or T5. The so-called monolingual models, such as **CamemBERT** [27] for French, **UmBERTo** [28] for Italian or **HerBERT** [29] for Polish to name a few, were trained on texts in a particular single language. In 2019, models called **XLM** [30] and **XLM-RoBERTa** [31] were presented, followed by **mT5** a year later [32]. These are examples of multilingual models, which are able to process and understand texts in more than 100 languages at once. If a business operates on texts in multiple languages, multilingual models may be the way to go.

The (hidden) cost of using an LLM

Training a large language model from scratch can be a very costly procedure (even hundreds of thousands or a few million US dollars in the case of the largest ones); it is no surprise, then, that companies frequently decide not to release their models for public use, or prefer to serve them via an API rather than sharing the model weights for free. An example of a model hidden behind an API is the GPT-3 and all newer LLMs provided by OpenAI, including the world-famous **ChatGPT** [33], with the cost of a fraction of a dollar for each 1000 tokens processed. Other API-based models (with similar pricing schemes) include **Jurassic-1** [34] and those offered by **Cohere** [35].

Soon after the release of GPT-3, unhappy with the model being “closed”, a group of independent scientists founded a non-profit organization called EleutherAI. Their goal was to reproduce GPT-3 to the greatest possible extent and release their model as an open-source alternative. Their efforts led to the consecutive development of **GPT-Neo** [36], **GPT-J** [37] and finally **GPT-NeoX** [38], with the last one reaching 20 billion parameters. Another initiative oriented towards open access to LLMs was the BigScience project, conducted in the form of a one-year-long research workshop, with 1000 researchers from over 250 institutions creating a multilingual language model called **BLOOM** [39] using a supercomputer located in France.

Researchers from Facebook (Meta) have also released two series of LLMs for public use: **OPT** [40] (mid-2022) and **LLaMA** [41] (February 2023), with the authors of the latter claiming that it beats larger alternatives on quality while being a few times smaller, aligned with the aforementioned Chinchilla scaling laws. LLaMA was trained using only publicly available datasets, but gave rise to some controversy, as unfortunately the model is licensed in such a way that only allows non-commercial applications, limiting their use to scientific research.

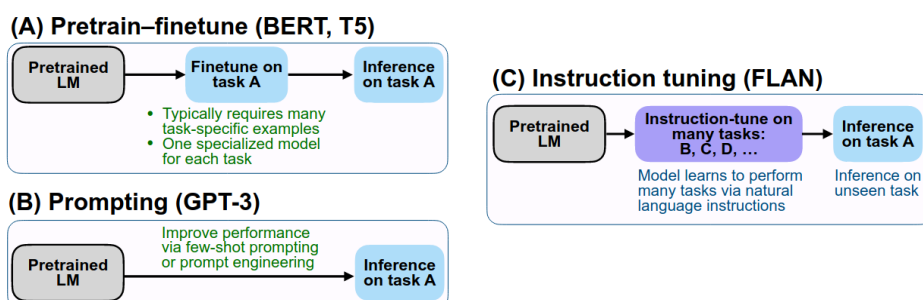
Obviously, even if the model weights are publicly available, using an LLM is not free of charge. It requires having the proper infrastructure for hosting the model, with specialized hardware (sufficiently powerful GPUs) being especially costly if one wants to use the largest models. Fortunately, for many tasks, even the smaller ones can be enough.

Emergent abilities of large language models

Not so long ago, the typical approach to training an NLP model to perform a specific task was the aforementioned transfer learning framework, in which a pre-trained language model is fine-tuned to new data in a supervised manner, i.e., by leveraging annotated corpora. However, the process of collecting annotations can often be time-consuming and expensive, slowing down or limiting the application of NLP to business problems. The situation has changed recently, with emergent abilities of large language models (a term coined in a mid-2022 paper [42]) coming to the rescue.

The pursuit of increasingly large models has led to an interesting discovery, first observed by OpenAI researchers in their GPT-3. As stated in the title of the related paper, at a certain scale, language models become capable of few-shot learning, which means they require only a few examples to be able to perform a previously unseen task with decent quality. In the case of GPT-3, the examples are provided to the model as part of the so-called prompt template. Despite the name, there is no “real” training, and the model just has to infer the answer from the input. The success of this prompting approach started discussions on prompt engineering becoming a profession in the future. As a side note, following the new research direction opened by the emergent few-shot performance of GPT-3, the vast majority of LLMs released afterwards are similar decoder-only architectures trained with an autoregressive objective.

Providing examples as part of the prompt template was a promising direction, but it was sometimes tricky and felt a bit cumbersome. At the same time, the zero-shot performance of GPT-3 (using prompts without examples) was rather unsatisfactory. In a paper published in late 2021, researchers from Google proposed their method for enabling zero-shot learning, called **FLAN** [43]. In this approach, the model learns to perform tasks via human-like natural language instructions rather than prompt templates, leading to much better zero-shot performance for previously unseen tasks. A year later, FLAN was also successfully applied to several models including T5 and PaLM, with **FLAN-T5** [44] being open-sourced and considered a powerful alternative to models hidden behind paid APIs. Examples of other instruction-tuned models that exhibit this zero-shot behavior include **T0** [45], **mT0** and **BLOOMZ** [46].



Comparison of fine-tuning, prompting and instruction tuning approaches.

Source: <https://arxiv.org/pdf/2109.01652.pdf>

At the beginning of 2022, natural language instructions were also used in yet another incarnation of the GPT model, the so-called **InstructGPT** [\[47\]](#) (known as the “text-davinci-003” model in the OpenAI API). Rather than relying purely on the language modeling objective, human-provided demonstrations of desired model behavior were used as part of the Reinforcement Learning from Human Feedback (RLHF) algorithm, which we will not dive deeper into in this text, as you can read more about it in a recent entry on our blog [\[48\]](#).

LLMs going mainstream

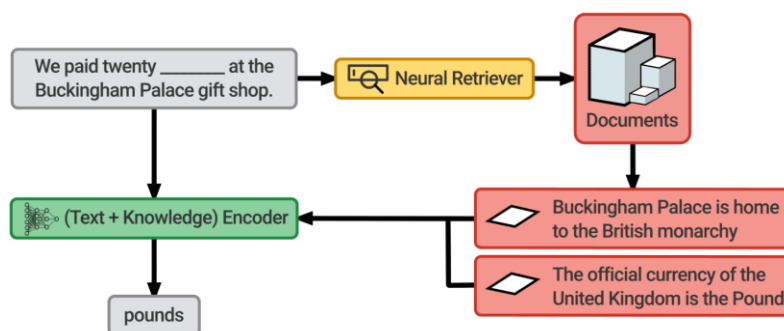
Recently, the **ChatGPT** model took the world by storm, reaching an unprecedented level of popularity and hype, especially among non-technical people. From the technical point of view, it is not so different from the InstructGPT described above. Apparently, what made people want to use it and led to its incredible success was the dialogue-oriented interface. You can read more details about ChatGPT in another recent blogpost [\[49\]](#).

After the enormously widespread adoption of ChatGPT including integration with Microsoft's Bing, Google announced they will soon respond with **Bard** [\[50\]](#), supposedly an improvement over their **LaMDA** model [\[51\]](#) from 2021, and make it part of their search engine. Who knows how the rivalry will end, especially if more companies join the competition. Other examples of dialogue-optimized LLMs include DeepMind's **Sparrow** [\[52\]](#) and Anthropic's assistant [\[53\]](#), referred to as **Claude**. For none of the chat-oriented LLMs are the weights publicly released.

Making LLMs know more than they already know

Although very impressive in terms of their text generation capabilities, LLMs are frequently accused of responding to questions with factually incorrect answers, and laughed at for being overly confident in their wrong responses. This limitation is actually expected, taking into account the way they are typically trained - they only possess the knowledge that was included in the training corpus.

To mitigate this issue, the research on so-called retrieval-augmented LLMs has recently been active. Such models use a “knowledge retriever” component, allowing you to find the answer to the provided question in a given collection of documents. The documents are first indexed, and then the content of the ones the retriever finds most relevant is added to the LLM input when generating the answer. The underlying mechanism of the retrieval-augmented models is based on ideas we have already discussed - **REALM** [54] (from Google) uses BERT for both retrieval and answer generation, **RAG** [55] (Facebook/Meta) takes advantage of BERT and BART in these two steps, while **ATLAS** [56] (also Facebook/Meta) makes use of T5 as a generator under the hood. Another model of this kind is **RETRO** [57] proposed by DeepMind, which, contrary to the abovementioned alternatives, is not publicly available.



A retrieval-augmented Large Language Model

Source: <https://ai.googleblog.com/2020/08/realm-integrating-retrieval-into.html>

GPT-4 – the future is now

OpenAI purposely marketed their InstructGPT and ChatGPT, both presented in 2022, as “GPT3.5 series” models, hinting that these were just preliminary steps on the path to the next, even more powerful version. Recent months were full of rumors suggesting that the **GPT-4** release is going to happen at some point in 2023. AI enthusiasts were speculating how large the model will be, with the guesstimated numbers reaching 100 trillion parameters.

The official announcement came mid-March, with a live demonstration, a blog post [\[58\]](#) and a technical report [\[59\]](#). Unfortunately, the report does not reveal any details about the model architecture (including model size) and hardware, with only limited information on the data and algorithms used for training. OpenAI researchers mention the competitive landscape and the safety implications of large-scale models as reasons behind the decision to keep these details undisclosed.

In the report, the claimed improvements over the preceding models are evidenced by evaluating GPT-4 on diverse datasets, including simulating exams that were originally designed for humans. On some of these, previously considered difficult, GPT-4 achieves human-level performance. What also makes GPT-4 distinct from all the models described above is that it is multimodal, which means it can handle not only text, but also visual input, allowing the users to ask questions about attached images or to extract knowledge from visually rich documents. At the time of writing, only the text functionality is made available via the OpenAI API (it requires signing up for the waitlist), with the image processing functionality still being improved before it gets fully launched.

Summary

The article was a long trek through the diverse landscape of Large Language Models. We learned about three main flavors: encoders, decoders and seq2seq models. We looked into the history of model scaling, from hundreds of millions to hundreds of billions of parameters. We emphasized the importance of data used for training, and mentioned the costs related to LLMs. Finally, we looked into interesting applications of these models. The pace at which new models and papers in the field of NLP emerge is extremely rapid. With the announcement of GPT-4 and new extensively researched ideas such as multimodal large vision-language models, the entire text may soon become outdated, but still, we hope you consider this text a valuable summary of what has been achieved in the field so far.

If you wish to find out more about the models described in this article (and many more), feel free to click on the link below: <https://github.com/azygadlo/LLM-catalog>

**Not sure how large language models
can boost your business?**

With our expertise and business mindset, we help you
to identify and implement practical solutions.

Contact us!

deepsense.ai
BIG DATA SCIENCE