Historique et contexte GN :

Définition :

# **Modèles de Transformers**

1. The original Transformer architecture is a specific instance of the encoder decoder models (Cho et al., 2014)2 that had become popular just over the 2–3 years prior. Up until that point however, attention was just one of the mechanisms used by these models, which were mostly based on LSTM (Long Short Term Memory) (Hochreiter & Schmidhuber, 1997) and other RNN (Recurrent Neural  
   Networks) (Mikolov et al., 2010) variations. The key insight of the Transformers  
   paper, as the title implies, was that attention could be used as the only mechanism  
   to derive dependencies between input and output.
2. In late 2017 the transformer architecture was first proposed by Vaswani et al. [[2017](https://dl.acm.org/doi/full/10.1145/3505245#Bib0177)]. At the time LSTMs were the prevailing paradigm in NLP, but they did not work terribly well or reliably for very long sequences or transfer learning. The transformer presented a novel way to incorporate an attention mechanism in deep feedforward networks that allowed it to capture long range sequence dependencies like the LSTM, but with a larger context window for longer sequences. The transformer was also easily parallelizable and highly scalable. Due to this, transformers have been trained using unprecedently large corpora [Raffel [2020](https://dl.acm.org/doi/full/10.1145/3505245#Bib0132); Brown [2020](https://dl.acm.org/doi/full/10.1145/3505245#Bib0022)]. We distinguish language models using the transformer as transformer language models (TLMs) because they perform remarkably better than LSTM-based models and they scale very well [Kaplan [2020](https://dl.acm.org/doi/full/10.1145/3505245#Bib0076)].
3. Un transformeur, ou modèle auto-attentif, est un modèle d'apprentissage profond introduit en 2017, utilisé principalement dans le domaine du traitement automatique des langues (TAL).
4. Les transformers sont des réseaux de neurones qui apprennent le contexte et la signification en suivant les relations dans les données séquentielles, comme les mots dans une phrase 1. Les transformers appliquent une série de techniques mathématiques évolutives, appelées attention ou auto-attention, pour détecter les subtilités des éléments de données distants dans une série qui influencent et dépendent les uns des autres 1.
5. Les Transformers sont rapidement devenus le modèle de choix pour les problèmes de TAL, remplaçant les anciens modèles de réseaux de neurones récurrents tels que le LSTM (Long Short-Term Memory) 2.
6. The Transformer in NLP is a novel architecture that aims to solve sequence-to-sequence tasks while handling long-range dependencies with ease. It relies entirely on self-attention to compute representations of its input and output WITHOUT using sequence-aligned RNNs or convolution.
7. The Transformer was proposed in the paper [Attention is All You Need](https://arxiv.org/abs/1706.03762). A TensorFlow implementation of it is available as a part of the [Tensor2Tensor](https://github.com/tensorflow/tensor2tensor) package. Harvard’s NLP group created a [guide annotating the paper with PyTorch implementation](http://nlp.seas.harvard.edu/2018/04/03/attention.html). In this post, we will attempt to oversimplify things a bit and introduce the concepts one by one to hopefully make it easier to understand to people without in-depth knowledge of the subject matter.
8. History

The history of [transformers](https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)) in neural networks can be traced back to the early 1990s, when Jürgen Schmidhuber proposed the first transformer model. This model was called the “fast weight controller” and it used a self-attention mechanism to learn the relationships between words in a sentence. However, the fast weight controller was not very efficient, and it was not widely used.

In 2017, Vaswani et al. published the paper “Attention is All You Need”, which introduced a new transformer model that was much more efficient than the fast weight controller. This new model, which is now simply called the “transformer”, quickly became state-of-the-art for a wide range of natural language efficient (NLP) tasks, including machine translation, text summarization, and question answering.

The transformer has been so successful because it is able to learn long-range dependencies between words in a sentence. This is essential for many NLP tasks, as it allows the model to understand the context of a word in a sentence. The transformer does this using a self-attention mechanism, which allows the model to focus on the most relevant words in a sentence when decoding the output tokens.

The transformer has had a major impact on the field of NLP. It is now the go-to approach for many NLP tasks, and it is constantly being improved. In the future, transformers are likely to be used to solve a wider range of NLP tasks, and they will become even more efficient and powerful.

Here are some of the key events in the history of transformers in neural networks:

* 1990: Jürgen Schmidhuber proposes the first transformer model, the “fast weight controller”.
* 2017: Vaswani et al. publish the paper “Attention is All You Need”, which introduces the transformer model.
* 2018: Transformer models achieve state-of-the-art results on a wide range of NLP tasks, including machine translation, text summarization, and question answering.
* 2019: Transformers are used to create large language models (LLMs) such as BERT and GPT-2.
* 2020: LLMs are used to create even more powerful models such as GPT-3.

The history of transformers in neural networks is still being written. It is an exciting time to be in the field of NLP, as transformers are making it possible to solve problems that were previously intractable.

The transformer model was introduced in 2017 by Google researchers in the paper "Attention is all you need" 2 3 . Before the transformer era, different AI architectures were predominant for different use cases, such as recurrent neural networks for language and convolutional neural networks for computer vision 1 . However, transformers have become the foundation of modern artificial intelligence and have revolutionized the field of AI over the past half-decade 1 3 . Virtually every advanced AI system is based on transformers, and every AI researcher is accustomed to working with them 1 . Transformers have been optimized by thousands of researchers building on one another's work over the past several years, giving them a powerful incumbency advantage that will make them formidable to dislodge 1 . The success of transformers has prompted researchers to explore their use in other AI areas, such as processing multiple kinds of input at once or planning tasks 3 . The transformer model has become so popular because it implies the potential to become universal, and researchers are exploring the possibility of a convergence of disparate subfields of AI 3 .

## Transformer model architecture

A transformer architecture consists of an encoder and decoder that work together. The attention mechanism lets transformers encode the meaning of words based on the estimated importance of other words or tokens. This enables transformers to process all words or tokens in parallel for faster performance, helping drive the growth of increasingly bigger LLMs.

Thanks to the attention mechanism, the encoder block transforms each word or token into vectors further weighted by other words. For example, in the following two sentences, the meaning of *it* would be weighted differently owing to the change of the word *filled* to *emptied*:

1. He poured the pitcher into the cup and filled it.
2. He poured the pitcher into the cup and emptied it.

The attention mechanism would connect *it* to the cup being filled in the first sentence and to the pitcher being emptied in the second sentence.

The decoder essentially reverses the process in the target domain. The original use case was translating English to French, but the same mechanism could translate short English questions and instructions into longer answers. Conversely, it could translate a longer article into a more concise summary.

Structure:

The transformer model architecture consists of two main blocks: the encoder and the decoder 1 4 6 . The encoder receives an input and builds a representation of it, while the decoder generates the output based on the encoder's representation 1 4 . The transformer model is composed of several blocks, each with its own function, including tokenization, embedding, positional encoding, and transformer blocks 1 6 . The transformer model uses self-attention layers to detect relationships between different elements in a sequence, allowing it to learn context and meaning by tracking relationships in sequential data like words in a sentence 1 5 . The attention mechanism lets transformers encode the meaning of words based on the estimated importance of other words or tokens 2 4 . The transformer model does not rely on recurrence and convolutions to generate an output sequence, unlike previous models 6 . The transformer model architecture is not that complex, and it is a concatenation of several blocks, each with its own function 6 . The transformer model has become the foundation of modern artificial intelligence and has revolutionized the field of AI over the past half-decade 3 .

The basic structure of a transformer model consists of two main blocks: the encoder and the decoder 1 4 6 . The encoder receives an input and builds a representation of it, while the decoder generates the output based on the encoder's representation 1 4 . The transformer model is composed of several blocks, each with its own function, including tokenization, embedding, positional encoding, and transformer blocks 1 6 . The transformer model uses self-attention layers to detect relationships between different elements in a sequence, allowing it to learn context and meaning by tracking relationships in sequential data like words in a sentence 1 5 . The attention mechanism lets transformers encode the meaning of words based on the estimated importance of other words or tokens 2 4 . The transformer model does not rely on recurrence and convolutions to generate an output sequence, unlike previous models 6 . The transformer model architecture is not that complex, and it is a concatenation of several blocks, each with its own function 6 .

The notion of attention has been around since the 1990s as a processing technique. However, in 2017 a team of Google workers suggested they could use attention to encode the meaning of words and the structure of a given language directly. This was revolutionary because it replaced what previously required an additional encoding step using a dedicated neural network. It also unlocked a way to virtually model any type of information, paving the way for the extraordinary breakthroughs that have emerged over the last several years.

Les transformateurs dans l'intelligence artificielle moderne:

Les transformateurs sont devenus la pierre angulaire de l'intelligence artificielle moderne, révolutionnant le paysage de l'IA au cours des cinq dernières années. Pratiquement tous les systèmes d'IA avancés reposent désormais sur des transformateurs, et les chercheurs se sont habitués à travailler avec eux. Les transformateurs sont difficiles à remplacer en raison de leur formidable avantage d'incumbance, qui est le résultat des efforts d'optimisation de milliers de chercheurs sur plusieurs années.

Application :

1. As a consequence, Transformer has become the go-to architecture in NLP, especially for PTMs. In addition to language related applications, Transformer has also been adopted in CV ([Parmar et al., 2018](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b94), [Carion et al., 2020](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b13), [Dosovitskiy et al., 2020](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b33)), [audio processing](https://www.sciencedirect.com/topics/computer-science/audio-signal-processing) ([Dong et al., 2018](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b32), [Gulati et al., 2020](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b41), [Chen et al., 2021](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b16)) and even other disciplines, such as chemistry ([Schwaller et al., 2019](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b114)) and life sciences ([Rives et al., 2021](https://www.sciencedirect.com/science/article/pii/S2666651022000146#b109)).

 Les Transformers sont également utilisés pour la vision par ordinateur, par exemple, la création des vision transformers (ViT) 2. Les transformers sont cruciaux dans toutes les applications de grands modèles de langage (LLM), y compris ChatGPT, Google Search, Dall-E et Microsoft Copilot [3. Les transformers peuvent être appliqués à pratiquement toutes les tâches qui traitent un type d'entrée donné pour générer une sortie, telles que la traduction d'une langue à une autre, la génération de chatbots plus engageants et utiles, etc.](https://www.techtarget.com/searchenterpriseai/definition/transformer-model" \t "_blank)

Emotion Detection (ED) is a branch of sentiment analysis (SA) that seeks to extract  
fine-grained emotions from either speech/voice, image, or text data. Detecting emotions  
from texts has suffered great paucity regardless of the quantum of text data available  
(Acheampong et al. 2020). This issue has been partly due to the absence of voice modulation, facial expressions, etc., which may present cues to aid in context and relation extraction. Another is partly due to the unavailability of a practical context extraction approach  
for texts. Besides, the need for emotion-conveying words disambiguation to verify classified emotions as real emotions presents a significant hitch in the field because some texts  
convey multiple emotional expressions. Recently, state of the art (SOTA) results are being  
obtained in the field using pre-trained transformer-based models

Modèles de Transformers Plus connus

1. In the last three years, hundreds of papers and language models inspired by Transformers were published, the bestknown being BERT [3], RoBERTa [4], AlBERT [5], XLNet [6], DistilBERT [7], and Reformer [8]. Some of the most popular Transformer models are included in the Transformers library, maintained by HuggingFace [9].
2. At the present moment, the dominant models for nearly all natural language processing tasks are based on the Transformer architecture. Given any new task in natural language processing, the default first-pass approach is to grab a large Transformer-based pretrained model, (e.g., BERT (Devlin et al., 2018), ELECTRA (Clark et al., 2020), RoBERTa (Liu et al., 2019), or Longformer (Beltagy et al., 2020)) adapting the output layers as necessary, and fine-tuning the model on the available data for the downstream task. If you have been paying attention to the last few years of breathless news coverage centered on OpenAI’s large language models, then you have been tracking a conversation centered on the GPT-2 and GPT-3 Transformer-based models (Brown et al., 2020, Radford et al., 2019). Meanwhile, the vision Transformer has emerged as a default model for diverse vision tasks, including image recognition, object detection, semantic segmentation, and superresolution (Dosovitskiy et al., 2021, Liu et al., 2021). Transformers also showed up as competitive methods for speech recognition (Gulati et al., 2020), reinforcement learning (Chen et al., 2021), and graph neural networks (Dwivedi and Bresson, 2020).

Les Transformers fournissent des milliers de modèles pré-entraînés pour effectuer des tâches de traitement sur des corpus de textes : la classification, l’extraction d’informations, la réponse aux questions, le résumé/la synthèse, la traduction, et ce dans une centaine de langues. Parmi ces modèles, les plus populaires sont BERT (Devlin et al., 2018), GPT v1-3 (Radford et al., 2018), RoBERTa (Liu et al., 2019) et T5 (Raffel et al., 2019).

Dans cette section, nous parlerons brièvement de certains modèles de Transformers populaires. Dans la section suivante, nous analyserons les modèles de grandes tailles qui nécessitent d’énormes capacités de calcul.

Le modèle GPT, Generative Pre-trained Transformer

Les Transformers génératifs pré-entraînés sont des modèles de langage développés (en 2018 pour GPT-3) par OpenAI. Ces modèles facilitent le développement d’applications d’apprentissage automatique tout en permettant aux personnes ayant peu d’expérience en tant que développeur informatique de créer des applications.

GPT-3 est l’un des modèles les plus populaires de cette famille. Il s’agit de la troisième génération des Transformers pré-entraînés génératifs, qui fonctionne avec environ 175 milliards de paramètres.

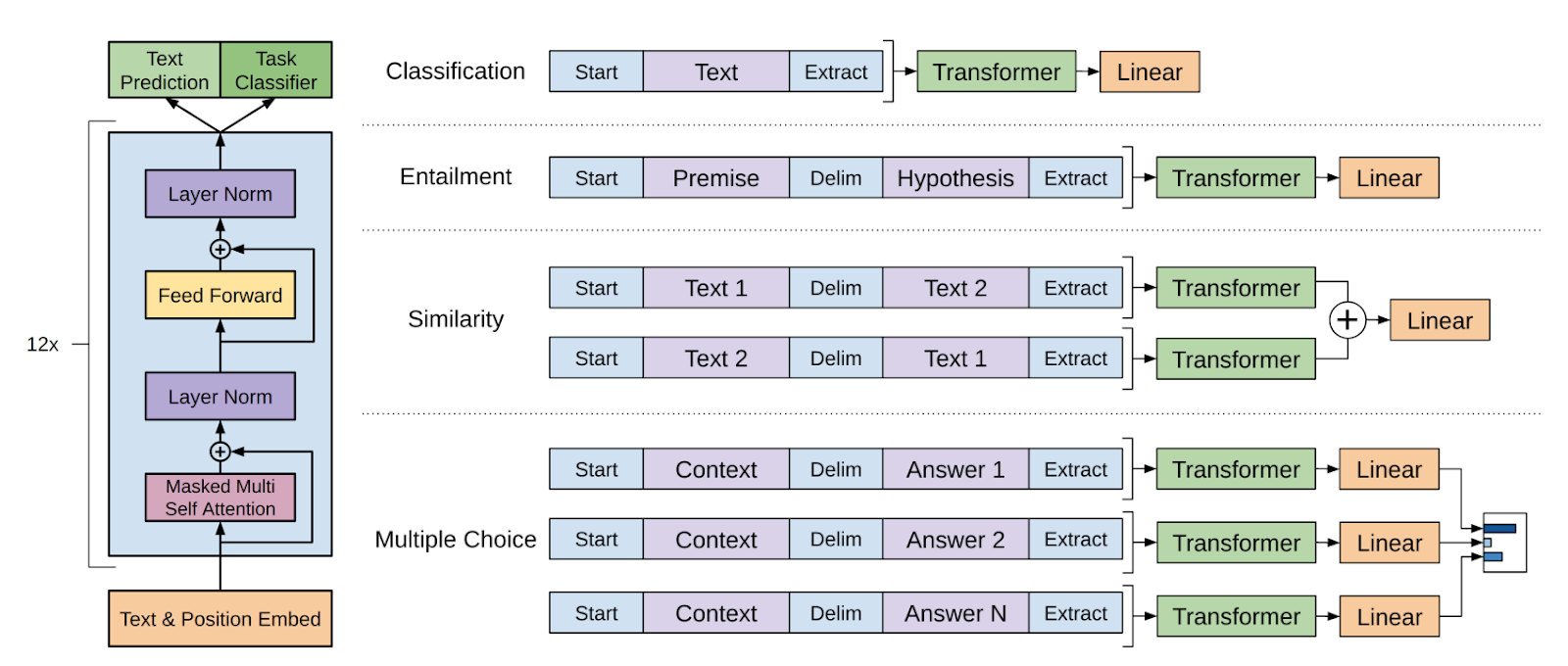


Figure 1 : (Gauche) Architecture générale du modèle GPT (Radford et al., 2018).  
(Droite) Transformations et différentes tâches (Radford et al., 2018)

Le modèle BERT, Bidirectionnel Encoder Representations from Transformers

BERT (Devlin et al., 2018) est un article publié par des chercheurs de Google AI. Il a fait sensation au sein de la communauté en présentant des résultats meilleurs pour une grande variété de tâches de NLP.

L’innovation technique clé de BERT consiste à appliquer l’entraînement bidirectionnel du Transformer. Contrairement aux modèles unidirectionnels, qui lisent le texte saisi de manière séquentielle (de gauche à droite ou de droite à gauche), l’encodeur du Transformer dans le modèle de BERT lit toute la séquence de mots en une fois. Cette caractéristique permet au modèle d’apprendre le contexte d’un mot en fonction de tout son environnement (gauche et droite du mot).

Il faut savoir que BERT est pré-entraîné sur deux tâches de NLP différentes mais liées : la modélisation du langage masqué (en anglais on parle de MLM, Masked Language Model) et la prédiction de la phrase suivante (NSP, Next Sentence Prediction). Le MLM fonctionne comme suit : l’algorithme commence par masquer un mot dans une phrase. Ensuite, l’algorithme va chercher à  prédire quels mots ont la plus grande probabilité d’être de bons remplaçants du mot caché, et ce en fonction du contexte. La NSP fonctionne en prédisant si deux phrases données ont une connexion logique (ou séquentielle) ou bien s’il n’y a pas de relation entre les phrases. Par exemple : Pierre est malade. Il a la grippe. Dans la dernière phrase, le pronom “il” renvoie à Pierre. Il y a donc une connexion logique causale entre ces deux phrases.

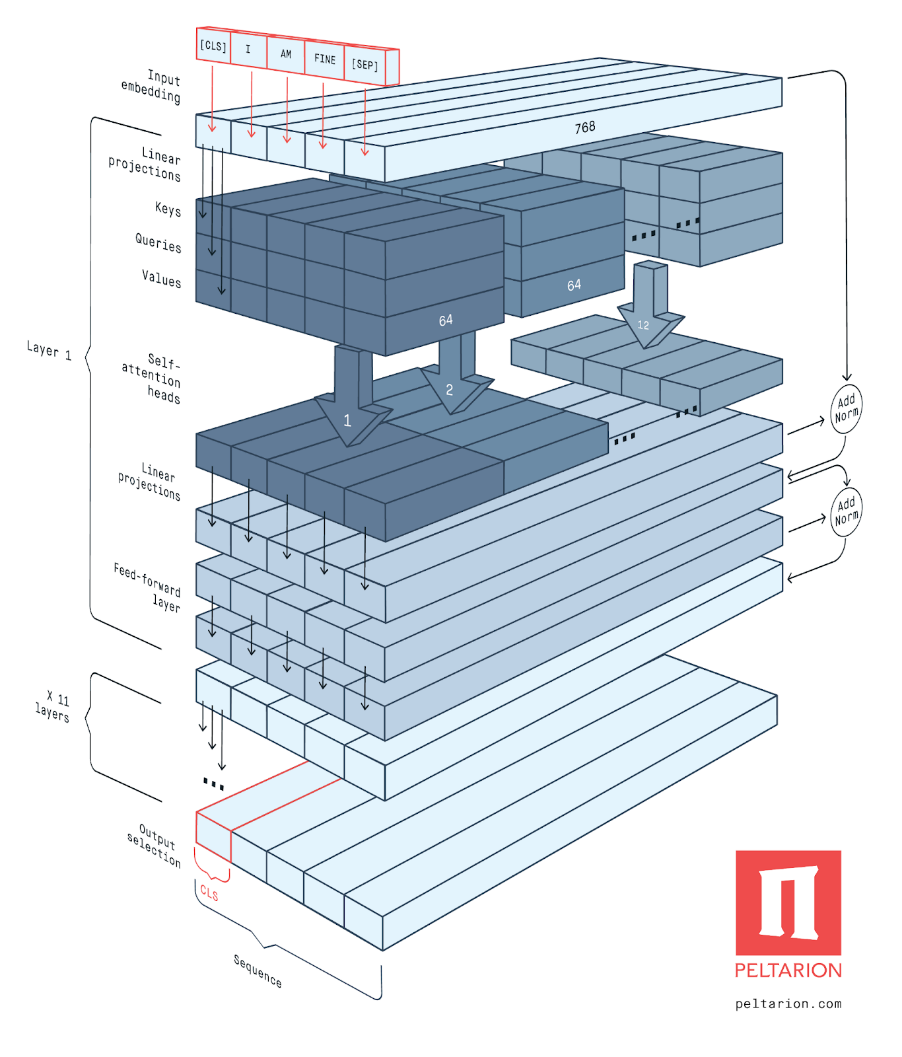


Figure 2 : Architecture générale de BERT (Peltarion, 2020)

RoBERTa

RoBERTa (Liu et al., 2019) s’appuie sur la stratégie de masquage MLM du modèle BERT, dans laquelle le système apprend à prédire des sections de texte intentionnellement cachées dans des exemples de langue non annotés. Au niveau technique, RoBERTa est une version de BERT pour laquelle certains hyperparamètres du pré-entraînement ont été modifiés. Pour RoBERTa, le méta-modèle NSP a été supprimé. De plus, en s’entraînant avec des mini-lots et des taux d’apprentissage beaucoup plus importants, RoBERTa est capable d’améliorer l’objectif de modélisation du langage masqué par rapport à BERT. Cela permet d’aboutir à de meilleures performances sur un grand nombre de tâches NLP.

CamemBERT

Les modèles de langage pré-entraînés sont désormais omniprésents dans le traitement du langage naturel. Malgré leurs succès, la plupart des modèles disponibles ont été formés soit sur des corpus textuels en langue anglaise, soit sur la concaténation de données dans plusieurs langues. Cela rend l’utilisation pratique de tels modèles limitée en France par exemple. Ainsi en France, CamemBERT a vu le jour. Il s’agit d’une version française des encodeurs bidirectionnels pour transformateurs (BERT).

CamemBERT (Martin et al., 2019) est un modèle linguistique de pointe pour le français, basé sur l’architecture RoBERTa pré-entraînée sur le corpus multilingue OSCAR. CamemBERT améliore l’état de l’art pour la plupart des tâches envisagées.

FlauBERT

Quelques mois après la sortie de CamemBERT, FlauBERT apparaît (Le et al., 2019). Il s’agit d’un modèle BERT français formé sur un corpus français très large et hétérogène. Des modèles de différentes tailles sont entraînés à l’aide du nouveau supercalculateur Jean Zay du CNRS (Centre National de la Recherche Scientifique). A noter que ce référentiel partage tous les modèles pré-entraînés, toutes les données, tous les codes-sources au grand public (si cela vous intéresse).

Avec FlauBERT vient FLUE : une configuration d’évaluation pour les systèmes NLP français similaire au célèbre benchmark GLUE. L’objectif est de permettre d’autres expériences reproductibles dans le futur et de partager des modèles pour la langue française. Les performances de FlauBERT et de CamemBERT sont très proches.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | BERT | RoBERTa | CamemBERT | FlauBERT |
| Langue | Anglais | Anglais | Français | Français |
| Données d’entraînement | 13 Go | 160 Go | 138 Go | 71 Go |
| Objectifs de pré- entraînement | NSP et MLM | MLM | MLM | MLM |
| Nombre de paramètres | 110 million | 125 million | 110 million | 138/373 million |
| Tokenizer | WordPiece 30K | BPE 50K | SentencePiece 32K | BPE 50K |
| Stratégie de masquage | Statique + Masquage de sous-mots | Dynamique + Masquage de sous-mots | Dynamique + Masquage de mots entiers | Dynamique + Masquage de sous-mots |

Figure 3 :  Comparaison entre les différentes variantes de BERT (Le et al., 2019)

Text-to-Text Transfer Transformer (T5)

T5 Transformer (Raffel et al., 2O19) est une architecture de Google basée sur le Transformer qui utilise une approche de texte à texte. Le principe d’entraînement est le suivant : chaque tâche est conçue pour alimenter le texte modèle en entrée et l’entraîner à générer du texte cible. Ce modèle est basé sur la théorie de l’apprentissage par transfert, où un modèle est d’abord pré-entraîné sur une tâche globale, riche en données avant d’être affiné sur une tâche plus spécifique.

T-NLG, Turing Natural Language Generation & MT-NLG, Megatron-Turing Natural Language Generation

T-NLG de Microsoft (Turing-NLG, 2020) est un modèle de langage génératif basé sur Transformer, ce qui signifie qu’il peut générer des mots pour accomplir des tâches textuelles ouvertes. En plus de compléter une phrase inachevée, il peut générer des réponses directes aux questions et des résumés de documents. T-NLG est un modèle de langage à 17 milliards de paramètres qui surpasse l’état de l’art sur de nombreuses tâches NLP courantes.

Dans le cadre d’un récent partenariat avec Microsoft, NVIDIA a présenté l’un des plus grands modèles de langage de Transformer, le modèle MT-NLG, Megatron-Turing Natural Language Generation à plus de 530 milliards de paramètres. Ce modèle est alimenté par les modèles de Transformer DeepSpeed et Megatron. L’entraînement de MT-NLG a été possible, grâce aux efforts combinés de NVIDIA et de Microsoft. Ces entreprises ont atteint une efficacité d’entraînement sans précédent en combinant l’infrastructure d’entraînement accélérée par GPU SOTA, avec une pile logicielle d’apprentissage distribuée.

Following the success of transformers in machine translation with the original sequence-to-sequence formulation, researchers began to explore their use in other NLP tasks. One of the most significant developments in this direction was the introduction of BERT (Bidirectional Encoder Representations from Transformers) by Devlin et al. [[2](https://www.mdpi.com/2078-2489/14/4/242#B2-information-14-00242)]. BERT is a pre-trained transformer model that can be fine-tuned for a wide range of NLP tasks, such as sentiment analysis, named entity recognition and question answering. BERT was trained using a contrastive task, where it was asked to predict missing tokens in a sentence given the context of the surrounding tokens. This approach allowed BERT to learn rich contextual representations of words, making it highly effective for a wide range of NLP tasks.

Another development in the field of transformers was the introduction of GPT (Generative Pretrained Transformer) by Radford et al. [[3](https://www.mdpi.com/2078-2489/14/4/242#B3-information-14-00242)]. GPT is a generative model trained on a large corpus of text with the goal of predicting the next token in a sequence given the context of the surrounding tokens. GPT has been shown to be highly effective for tasks such as text generation, language modeling, and question-answering. Unlike BERT, which was trained using a contrastive task, GPT was trained using a generative task, allowing it to learn a more diverse and complete representation of language.

## Quelles sont les applications des Transformers ?

Même si à l’origine les Transformers ont été designés pour le traitement du langage, leurs performances les ont rendus utiles dans de nombreux domaines.

Ils ont largement été introduits pour de nombreuses tâches comme :

* Traitement du langage naturel (NLP): Les transformers sont souvent utilisés pour les tâches NLP telles que la classification de texte, la génération de texte, la traduction automatique, etc.
* Recommandation: Les modèles de transformers peuvent être utilisés pour recommander des produits ou des contenus à des utilisateurs en fonction de leur historique d’achats ou de navigation.
* Vision par ordinateur: Les transformers peuvent être utilisés pour les tâches de vision par ordinateur, telles que la reconnaissance d’objets, la génération d’images et la segmentation d’images.
* Jeux: Les transformers peuvent être utilisés pour les jeux, tels que les jeux d’aventure textuels, en fournissant des réponses contextuelles à des joueurs.
* Finance: Les transformers peuvent être utilisés pour les applications financières telles que la classification de données financières et la prédiction des tendances du marché.

### Transformer Language Models (TLMs)

From sentiment analysis to emotion detection to text classification to regression to cross-lingual analysis, TLMs promise to have a significant positive impact on future IS research, notwithstanding the more advanced novel applications, and they can do so in a number of different ways. For improving existing work, they can either (1) be used to generate rich CWRs or (2) be used directly, through pretraining, through fine-tuning and DTL, or both. More exciting, they can (3) be used to extend existing IS research by enabling easier cross-cultural analyses. We describe these themes in this subsection and discuss other issues that may impact the future use of TLMs.

### Les mécanismes d'attention

Les mécanismes d'attention sont les moyens de faire passer au décodeur l'information de quelles étapes de l'encodeur (i.e. quels mots de la séquence d'entrée) sont les plus importantes au moment de générer un mot de sortie. Quels sont les mots de la séquence d'entrée qui se rapportent le plus au mot qu'il est en train de générer en sortie, soit qu'ils s'y rapportent comme contexte pour lui donner un sens, soit qu'ils s'y rapportent comme mots "cousins" de signification proche.

**Les mécanismes d'auto-attention**(self-attention) sont similaires, sauf qu'**au lieu de s'opérer entre les éléments de l'encodeur et du décodeur**, ils **s'opèrent sur les éléments de l'input entre eux** (le présent regarde le passé et le futur) **et de l'output entre eux** aussi (le présent regarde le passé, vu que le futur est encore à générer).

Par exemple au moment de générer "feel", c'est en fait l'ensemble "avoir la pêche" qui a du sens, il doit faire attention à tous les mots, et idem quand il génère "great", car il s'agit de la traduction d'une expression idiomatique

The final language model we mention here is also a generative language model: the **generative pretrained transformer 3 (GPT-3)** [Brown [2020](https://dl.acm.org/doi/full/10.1145/3505245#Bib0022)]. GPT-3 is an OpenAI TLM which uses the same architecture as its predecessor, GPT-2 [Radford [2019](https://dl.acm.org/doi/full/10.1145/3505245#Bib0131)]. What makes GPT-3 unique is its scale: GPT-3 was scaled to a model size, measured by number of parameters, an order of magnitude larger than any previous model and was pretrained on the largest dataset to date. This required extreme investments in computational resources and distributed computing infrastructure, but led to surprising improvements on zero-, one- and few-shot learning tasks.