

Analysis of the evolution of advanced transformer-based language models: experiments on opinion mining

Nour Eddine Zekaoui, Siham Yousfi, Maryem Rhanoui, Mounia Mikram

Meridian Team, LYRICA Laboratory, School of Information Sciences, Rabat, Morocco

Article Info

Article history:

Received Jan 5, 2023

Revised Jan 16, 2023

Accepted Mar 10, 2023

Keywords:

Natural language processing

Opinion mining

Transformer-based models

ABSTRACT

Opinion mining, also known as sentiment analysis, is a subfield of natural language processing (NLP) that focuses on identifying and extracting subjective information in textual material. This can include determining the overall sentiment of a piece of text (e.g., positive or negative), as well as identifying specific emotions or opinions expressed in the text, that involves the use of advanced machine and deep learning techniques. Recently, transformer-based language models make this task of human emotion analysis intuitive, thanks to the attention mechanism and parallel computation. These advantages make such models very powerful on linguistic tasks, unlike recurrent neural networks that spend a lot of time on sequential processing, making them prone to fail when it comes to processing long text. The scope of our paper aims to study the behaviour of the cutting-edge Transformer-based language models on opinion mining and provide a high-level comparison between them to highlight their key particularities. Additionally, our comparative study shows leads and paves the way for production engineers regarding the approach to focus on and is useful for researchers as it provides guidelines for future research subjects.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Nour Eddine Zekaoui

Meridian Team, LYRICA Laboratory, School of Information Sciences

Rabat, Morocco

Email: noureddinezekaoui@gmail.com, nour-eddine.zekaoui@esi.ac.ma

1. INTRODUCTION

Over the past few years, interest in natural language processing (NLP) [1] has increased significantly. Today, several applications are investing massively in this new technology, such as extending recommender systems [2], [3], uncovering new insights in the health industry [4], [5], and unraveling e-reputation and opinion mining [6], [7]. Opinion mining is an approach to computational linguistics and NLP that automatically identifies the emotional tone, sentiment, or thoughts behind a body of text. As a result, it plays a vital role in driving business decisions in many industries. However, seeking customer satisfaction is costly expensive. Indeed, mining user feedback regarding the products offered, is the most accurate way to adapt strategies and future business plans. In recent years, opinion mining has seen considerable progress, with applications in social media and review websites. Recommendation may be staff-oriented [2] or user-oriented [8] and should be tailored to meet customer needs and behaviors.

Nowadays, analyzing people's emotions has become more intuitive thanks to the availability of many large pre-trained language models such as bidirectional encoder representations from transformers (BERT) [9] and its variants. These models use the seminal transformer architecture [10], which is based solely on attention mechanisms, to build robust language models for a variety of semantic tasks, including text classification.

Moreover, there has been a surge in opinion mining text datasets, specifically designed to challenge NLP models and enhance their performance. These datasets are aimed at enabling models to imitate or even exceed human level performance, while introducing more complex features.

Even though many papers have addressed NLP topics for opinion mining using high-performance deep learning models, it is still challenging to determine their performance concretely and accurately due to variations in technical environments and datasets. Therefore, to address these issues, our paper aims to study the behaviour of the cutting-edge transformer-based models on textual material and reveal their differences. Although, it focuses on applying both transformer encoders and decoders, such as BERT [9] and generative pre-trained transformer (GPT) [11], respectively, and their improvements on a benchmark dataset. This enable a credible assessment of their performance and understanding their advantages, allowing subject matter experts to clearly rank the models. Furthermore, through ablations, we show the impact of configuration choices on the final results.

2. BACKGROUND

2.1. Transformer

The transformer [10], as illustrated in Figure 1, is an encoder-decoder model dispensing entirely with recurrence and convolutions. Instead, it leverages the attention mechanism to compute high-level contextualized embeddings. Being the first model to rely solely on attention mechanisms, it is able to address the issues commonly associated with recurrent neural networks, which factor computation along symbol positions of input and output sequences, and then precludes parallelization within samples. Despite this, the transformer is highly parallelizable and requires significantly less time to train. In the upcoming sections, we will highlight the recent breakthroughs in NLP involving transformer that changed the field overnight by introducing its designs, such as BERT [9] and its improvements.

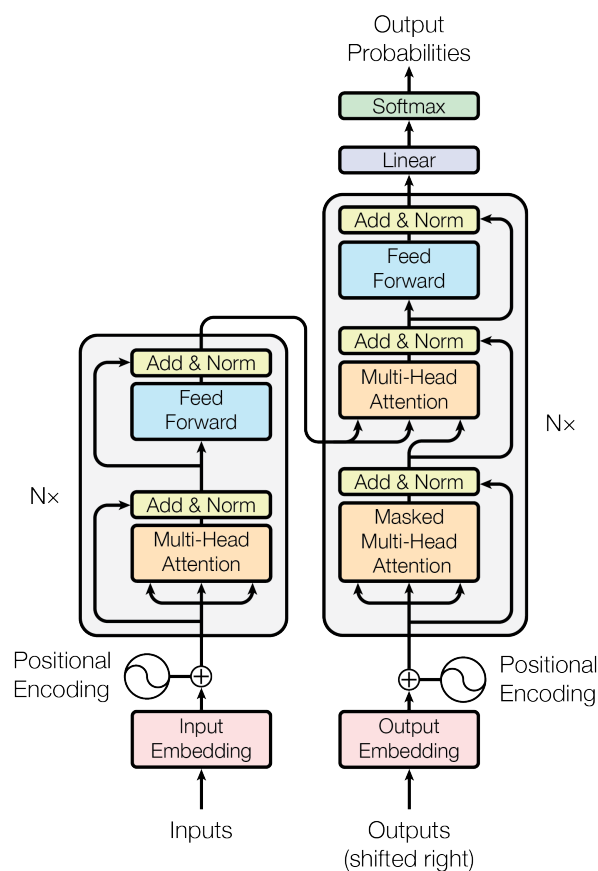


Figure 1. The transformer model architecture [10]

2.2. BERT

BERT [9] is pre-trained using a combination of masked language modeling (MLM) and next sentence prediction (NSP) objectives. It provides high-level contextualized embeddings grasping the meaning of words in different contexts through global attention. As a result, the pre-trained BERT model can be fine-tuned for a wide range of downstream tasks, such as question answering and text classification, without substantial task-specific architecture modifications.

BERT and its variants allow the training of modern data-intensive models. Moreover, they are able to capture the contextual meaning of each piece of text in a way that traditional language models are unfit to do, while being quicker to develop and yielding better results with less data. On the other hand, BERT and other large neural language models are very expensive and computationally intensive to train/fine-tune and make inference.

2.3. GPT-I, II, III

GPT [11] is the first causal or autoregressive transformer-based model pre-trained using language modeling on a large corpus with long-range dependencies. However, its bigger and optimized version called GPT-2 [12], was pre-trained on WebText. Likewise, GPT-3 [13] is architecturally similar to its predecessors. Its higher level of accuracy is attributed to its increased capacity and greater number of parameters, and it was pre-trained on Common Crawl. The OpenAI GPT family models has taken pre-trained language models by storm, they are very powerful on realistic human text generation and many other miscellaneous NLP tasks. Therefore, a small amount of input text can be used to generate large amount of high-quality text, while maintaining semantic and syntactic understanding of each word.

2.4. ALBERT

A lite BERT (ALBERT) [14] was proposed to address the problems associated with large models. It was specifically designed to provide contextualized natural language representations to improve the results on downstream tasks. However, increasing the model size to pre-train embeddings becomes harder due to memory limitations and longer training time. For this reason, this model arose.

ALBERT is a lighter version of BERT, in which next sentence prediction (NSP) is replaced by sentence order prediction (SOP). In addition to that, it employs two parameter-reduction techniques to reduce memory consumption and improve training time of BERT without hurting performance:

- Splitting the embedding matrix into two smaller matrices to easily grow the hidden size with fewer parameters, ALBERT separates the hidden layers size from the size of the vocabulary embedding by decomposing the embedding matrix of the vocabulary.
- Repeating layers split among groups to prevent the parameter from growing with the depth of the network.

2.5. RoBERTa

The choice of language model hyper-parameters has a substantial impact on the final results. Hence, robustly optimized BERT pre-training approach (RoBERTa) [15] is introduced to investigate the impact of many key hyper-parameters along with data size on model performance. RoBERTa is based on Google's BERT [9] model and modifies key hyper-parameters, where the masked language modeling objective is dynamic and the NSP objective is removed. It is an improved version of BERT, pre-trained with much larger mini-batches and learning rates on a large corpus using self-supervised learning.

2.6. XLNet

The bidirectional property of transformer encoders, such as BERT [9], help them achieve better performance than autoregressive language modeling based approaches. Nevertheless, BERT ignores dependency between the positions masked, and suffers from a pretrain-finetune discrepancy when relying on corrupting the input with masks. In view of these pros and cons, XLNet [16] has been proposed. XLNet is a generalized autoregressive pretraining approach that allows learning bidirectional dependencies by maximizing the anticipated likelihood over all permutations of the factorization order. Furthermore, it overcomes the drawbacks of BERT [9] due to its casual or autoregressive formulation, inspired from the transformer-XL [17].

2.7. DistilBERT

Unfortunately, the outstanding performance that comes with large-scale pretrained models is not cheap. In fact, operating them on edge devices under constrained computational training or inference budgets remains challenging. Against this backdrop, DistilBERT [18] (or Distilled BERT) has seen the light to address the cited issues by leveraging knowledge distillation [19].

DistilBERT is similar to BERT, but it is smaller, faster, and cheaper. It has 40% less parameters than BERT base, runs 40% faster, while preserving over 95% of BERT's performance. It is trained using distillation of the pretrained BERT base model.

2.8. XLM-RoBERTa

Pre-trained multilingual models at scale, such as multilingual BERT (mBERT) [9] and cross-lingual language models (XLMs) [20], have led to considerable performance improvements for a wide variety of cross-lingual transfer tasks, including question answering, sequence labeling, and classification. However, the multilingual version of RoBERTa [15] called XLM-RoBERTa [21], pre-trained on the newly created 2.5TB multilingual CommonCrawl corpus containing 100 different languages, has further pushed the performance. It has shown strong improvements on low-resource languages compared to previous multilingual models.

2.9. BART

Bidirectional and auto-regressive transformer (BART) [22] is a generalization of BERT [9] and GPT [11], it takes advantage of the standard transformer [10]. Concretely, it uses a bidirectional encoder and a left-to-right decoder. It is trained by corrupting text with an arbitrary noising function and learning a model to reconstruct the original text. BART has shown phenomenal success when fine-tuned on text generation tasks such as translation, but also performs well for comprehension tasks like question answering and classification.

2.10. ConvBERT

While BERT [9] and its variants have recently achieved incredible performance gains in many NLP tasks compared to previous models, BERT suffers from large computation cost and memory footprint due to reliance on the global self-attention block. Although all its attention heads, BERT was found to be computationally redundant, since some heads simply need to learn local dependencies. Therefore, ConvBERT [23] is a better version of BERT [9], where self-attention blocks are replaced with new mixed ones that leverage convolutions to better model global and local context.

2.11. Reformer

Consistently, large transformer [10] models achieve state-of-the-art results in a large variety of linguistic tasks, but training them on long sequences is costly challenging. To address this issue, the Reformer [24] was introduced to improve the efficiency of transformers while holding the high performance and the smooth training. Reformer is more efficient than transformer [10] thanks to locality-sensitive hashing attention and reversible residual layers instead of the standard residuals, and axial position encoding and other optimizations.

2.12. T5

Transfer learning has emerged as one of the most influential techniques in NLP. Its efficiency in transferring knowledge to downstream tasks through fine-tuning has given birth to a range of innovative approaches. One of these approaches is transfer learning with a unified text-to-text transformer (T5) [25], which consists of a bidirectional encoder and a left-to-right decoder. This approach is reshaping the transfer learning landscape by leveraging the power of being pre-trained on a combination of unsupervised and supervised tasks and reframing every NLP task into text-to-text format.

2.13. ELECTRA

Masked language modeling (MLM) approaches like BERT [9] have proven to be effective when transferred to downstream NLP tasks, although, they are expensive and require large amounts of compute. Efficiently learn an encoder that classifies token replacements accurately (ELECTRA) [26] is a new pre-training approach that aims to overcome these computation problems by training two Transformer models: the generator and the discriminator. ELECTRA trains on a replaced token detection objective, using the discriminator to identify which tokens were replaced by the generator in the sequences. Unlike MLM-based models, ELECTRA is defined over all input tokens rather than just a small subset that was masked, making it a more efficient pre-training approach.

2.14. Longformer

While previous transformers were focusing on making changes to the pre-training methods, the long-document transformer (Longformer) [27] comes to change the transformer's self-attention mechanism. It has become the de facto standard for tackling a wide range of complex NLP tasks, with a new attention mechanism that scales linearly with sequence length, and then being able to easily process longer sequences. Longformer's new attention mechanism is a drop-in replacement for the standard self-attention and combines a local windowed attention with a task motivated global attention. Simply, it replaces the transformer [10] attention matrices with sparse matrices for higher training efficiency.

2.15. DeBERTa

DeBERTa [28] stands for decoding-enhanced BERT with disentangled attention. It is a pre-training approach that extends Google's BERT [9] and builds on the RoBERTa [15]. Despite being trained on only half of the data used for RoBERTa, DeBERTa has been able to improve the efficiency of pre-trained models through the use of two novel techniques:

- Disentangled attention (DA): an attention mechanism that computes the attention weights among words using disentangled matrices based on two vectors that encode the content and the relative position of each word respectively.
- Enhanced mask decoder (EMD): a pre-trained technique used to replace the output softmax layer. Thus, incorporate absolute positions in the decoding layer to predict masked tokens for model pre-training.

3. APPROACH

Transformer-based pre-trained language models have led to substantial performance gains, but careful comparison between different approaches is challenging. Therefore, we extend our study to uncover insights regarding their fine-tuning process and main characteristics. Our paper first aims to study the behavior of these models, following two approaches: a data-centric view focusing on the data state and quality, and a model-centric view giving more attention to the models tweaks. Indeed, we will see how data processing affects their performance and how adjustments and improvements made to the model over time is changing its performance. Thus, we seek to end with some takeaways regarding the optimal setup that aids in cross-validating a Transformer-based model, specifically model tuning hyper-parameters and data quality.

3.1. Models summary

In this section, we present the base versions' details of the models introduced previously as shown in Table A1. We aim to provide a fair comparison based on the following criteria: L-Number of transformer layers, H-Hidden state size or model dimension, A-Number of attention heads, number of total parameters, tokenization algorithm, data used for pre-training, training devices and computational cost, training objectives, good performance tasks, and a short description regarding the model key points [29]. All these information will help to understand the performance and behaviors of different transformer-based models and aid to make the appropriate choice depending on the task and resources.

3.2. Configuration

It should be noted that we have used almost the same architecture building blocks for all our implemented models as shown in Figure 2 and Figure 3 for both encoder and decoder based models, respectively. In contrast, seq2seq models like BART are merely a bidirectional encoder pursued by an autoregressive decoder. Each model is fed with the three required inputs, namely input ids, token type ids, and attention mask. However, for some models, the position embeddings are optional and can sometimes be completely ignored (e.g RoBERTa), for this reason we have blurred them a bit in the figures. Furthermore, it is important to note that we uniformed the dataset in lower cases, and we tokenized it with tokenizers based on WordPiece [30], SentencePiece [31], and Byte-pair-encoding [32] algorithms.

In our experiments, we used a highly optimized setup using only the base version of each pre-trained language model. For training and validation, we set a batch size of 8 and 4, respectively, and fine-tuned the models for 4 epochs over the data with maximum sequence length of 384 for the intent of correspondence to the majority of reviews' lengths and computational capabilities. The AdamW optimizer is utilized to optimize the models with a learning rate of 3e-5 and the epsilon (eps) used to improve numerical stability is set to 1e-6, which is the default value. Furthermore, the weight decay is set to 0.001, while excluding bias, LayerNorm.bias,

and LayerNorm.weight from the decay weight when fine-tuning, and not decaying them when it is set to 0.000. We implemented all of our models using PyTorch and transformers library from Hugging Face, and ran them on an NVIDIA Tesla P100-PCIE GPU-Persistence-M (51G) GPU RAM.

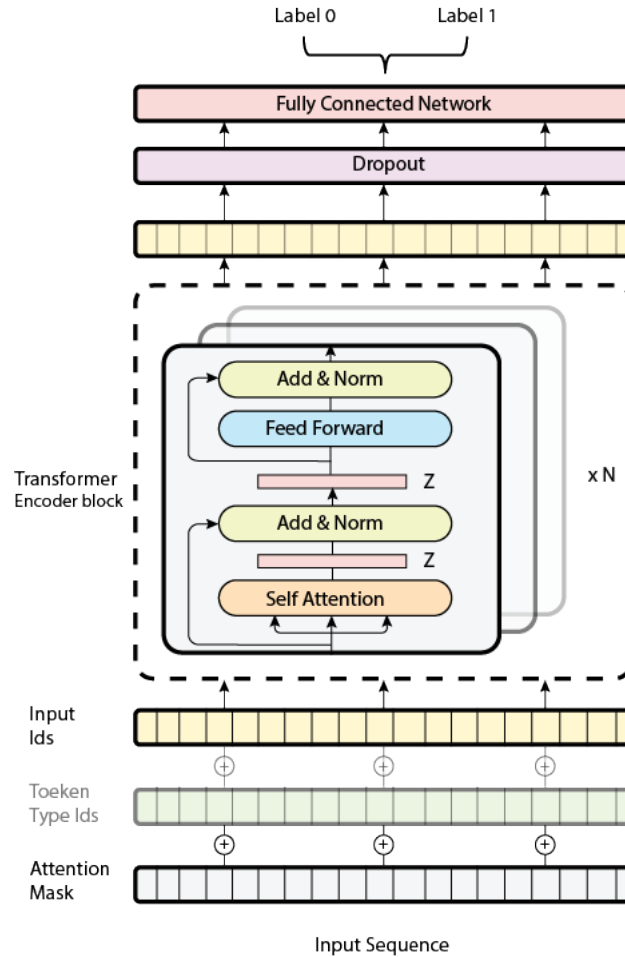


Figure 2. The architecture of the transformer encoder-based models

3.3. Evaluation

Dataset to fine-tune our models, we used the IMDb movie review dataset [33]. A binary sentiment classification dataset having 50K highly polar movie reviews labelled in a balanced way between positive and negative. We chose it for our study because it is often used in research studies and is a very popular resource for researchers working on NLP and ML tasks, particularly those related to sentiment analysis and text classification due to its accessibility, size, balance and pre-processing. In other words, it is easily accessible and widely available, with over 50K reviews well-balanced, with an equal number of positive and negative reviews as shown in Figure 4. This helps prevent biases in the trained model. Additionally, it has already been pre-processed with the text of each review cleaned and normalized.

Metrics to assess the performance of the fine-tuned transformers on the IMDb movie reviews dataset, tracking the loss and accuracy learning curves for each model is an effective method. These curves can help detect incorrect predictions and potential overfitting, which are crucial factors to consider in the evaluation process. Moreover, widely-used metrics, namely accuracy, recall, precision, and F1-score are valuable to consider when dealing with classification problems. These metrics can be defined as:

$$Precision = \frac{TP}{TP + FP} \quad , \quad Recall = \frac{TP}{TP + FN} \quad , \text{ and } F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (1)$$

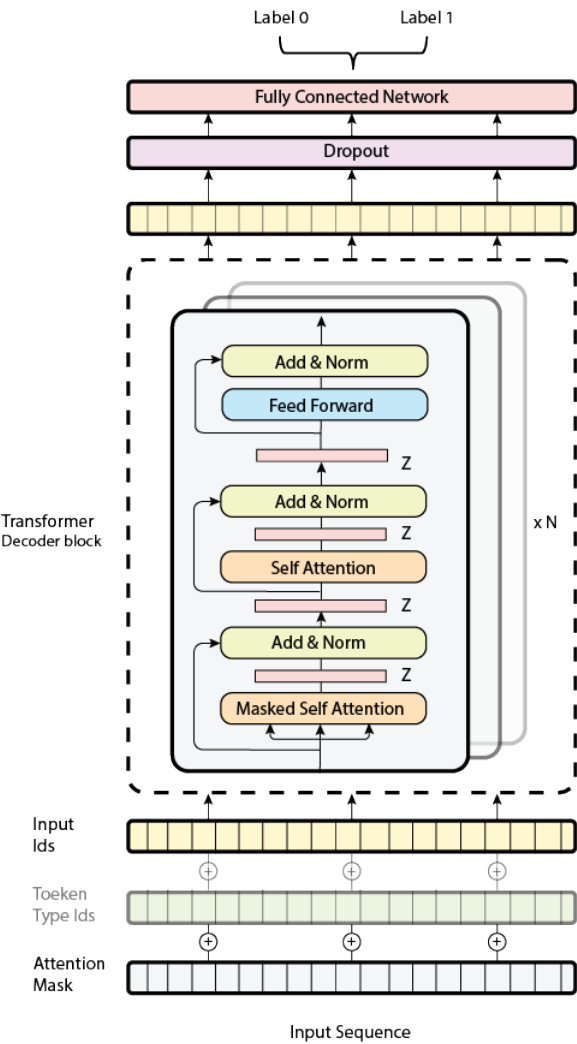


Figure 3. The architecture of the transformer decoder-based models

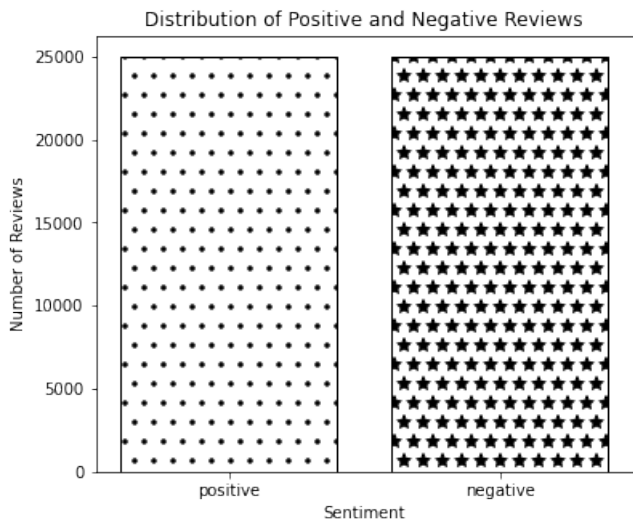


Figure 4. Positive and negative reviews distribution

4. RESULTS

In this section, we present the fine-tuning main results of our implemented transformer-based language models on the opinion mining task on the IMDb movie reviews dataset. Typically, all the fine-tuned models perform well with fairly high performance, except the three autoregressive models: GPT, GPT-2, and Reformer, as shown in Table 1. The best model, ELECTRA, provides an F1-score of 95.6 points, followed by RoBERTa, Longformer, and DeBERTa, with an F1-score of 95.3, 95.1, and 95.1 points, respectively. On the other hand, the worst model, GPT-2 provide an F1-score of 52.9 points as shown in Figure 5 and Figure 6. From the results, it is clear that purely autoregressive models do not perform well on comprehension tasks like sentiment classification, where sequences may require access to bidirectional contexts for better word representation, therefore, good classification accuracy. Whereas, with autoencoding models taking advantage of left and right contexts, we saw good performance gains. For instance, the autoregressive XLNet model is our fourth best model in Table 1 with an F1 score of 94.9%, it incorporates modelling techniques from autoencoding models into autoregressive models while avoiding and addressing limitations of encoders. The code and fine-tuned models are available at [34].

Table 1. Transformer-based language models validation performance on the opinion mining IMDb dataset

Model	Recall	Precision	F1	Accuracy
BERT	93.9	94.3	94.1	94.0
GPT	92.4	51.8	66.4	53.2
GPT-2	51.1	54.8	52.9	54.5
ALBERT	94.1	91.9	93.0	93.0
RoBERTa	96.0	94.6	95.3	95.3
XLNet	94.7	95.1	94.9	94.8
DistilBERT	94.3	92.7	93.5	93.4
XLM-RoBERTA	83.1	71.7	77.0	75.2
BART	96.0	93.3	94.6	94.6
ConvBERT	95.5	93.7	94.6	94.5
DeBERTa	95.2	95.0	95.1	95.1
ELECTRA	95.8	95.4	95.6	95.6
Longformer	95.9	94.3	95.1	95.0
Reformer	54.6	52.1	53.3	52.2
T5	94.8	93.4	94.0	93.9

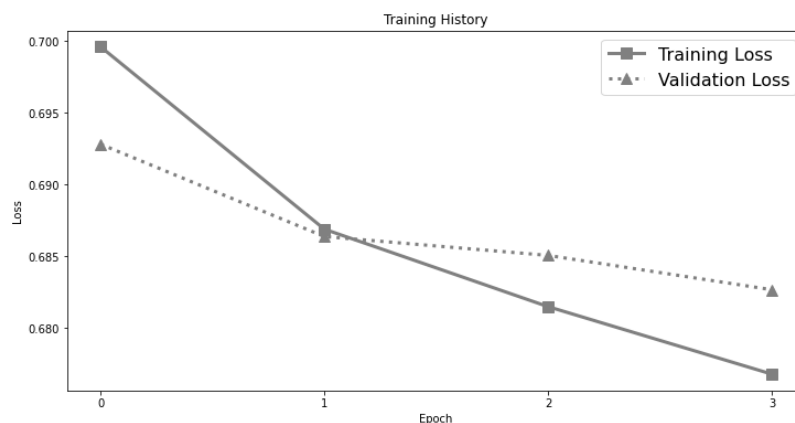


Figure 5. Worst model: GPT-2 loss learning curve

5. ABLATION STUDY

In Table 2 and Figure 7, we demonstrate the importance of configuration choices through controlled trials and ablation experiments. Indeed, the maximum length of the sequence and data cleaning are particularly crucial. Thus, to make our ablation study credible, we fine-tuned our BERT model with the same setup, changing only the sequence length (max-len) initially and cleaning the data (cd) at another time to observe how they affect the performance of the model.

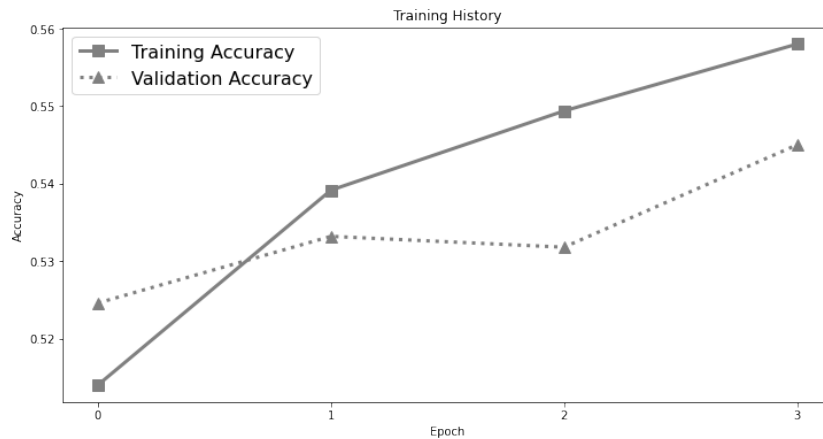


Figure 6. Worst model: GPT-2 acc learning curve

Table 2. Validation results of the BERT model based on different configurations, where cd stands for cleaned data, meaning that the latest model (BERT_{max-len=384, cd}) is trained on an exhaustively cleaned text

Model	Recall	Precision	F1	Accuracy
BERT _{max-len=64}	86.8%	84.7%	85.8%	85.6%
BERT _{max-len=384}	93.9%	94.3%	94.1%	94.0%
BERT _{max-len=384, cd}	92.6%	91.6%	92.1%	92.2%

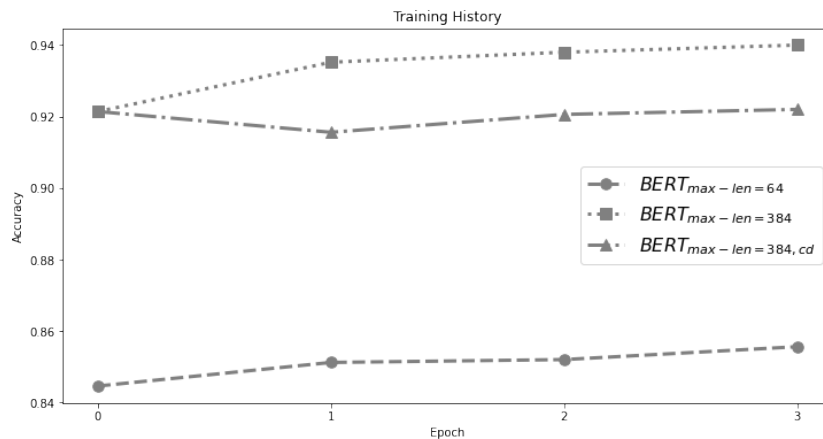


Figure 7. Validation accuracy history of BERT model based on different configurations

5.1. Effects of hyper-parameters

The gap between the performance of BERT_{max-len=64} and BERT_{max-len=384} on the IMDb dataset is an astounding 8.3 F1 points, as in Table 2, demonstrating how important this parameter is. Thereby, visualizing the distribution of tokens or words count is the ultimate solution for defining the optimal and correct value of the maximum length parameter that corresponds to all the training data points. Figure 8 illustrates the distribution of the number of tokens in the IMDb movie reviews dataset, it shows that the majority of reviews are between 100 and 400 tokens in length. In this context, we chose 384 as the maximum length reference to study the effect of the maximum length parameter, because it covers the majority of review lengths while conserving memory and saving computational resources. It should be noted that the BERT model can process texts up to 512 tokens in length. It is a consequence of the model architecture and can not be adjusted directly.

5.2. Effects of data cleaning

Traditional machine learning algorithms require extensive data cleaning before vectorizing the input sequence and then feeding it to the model, with the aim of improving both reliability and quality of the data.

Therefore, the model can only focus on important features during training. Contrarily, the performance dropped down dramatically by 2 F1 points when we cleaned the data for the BERT model. Indeed, the cleaning carried out aims to normalize the words of each review. It includes lemmatization to group together the different forms of the same word, stemming to reduce a word to its root, which is affixed to suffixes and prefixes, deletion of URLs, punctuations, and patterns that do not contribute to the sentiment, as well as the elimination of all stop words, except the words “no”, “nor”, and “not”, because their contribution to the sentiment can be tricky. For instance, “Black Panther is boring” is a negative review, but “Black Panther is not boring” is a positive review. This drop can be justified by the fact that BERT model and attention-based models need all the sequence words to better capture the meaning of words’ contexts. However, with cleaning, words may be represented differently from their meaning in the original sequence. Note that “not boring” and “boring” are completely different in meaning, but if the stop word “not” is removed, we end up with two similar sequences, which is not good in sentiment analysis context.

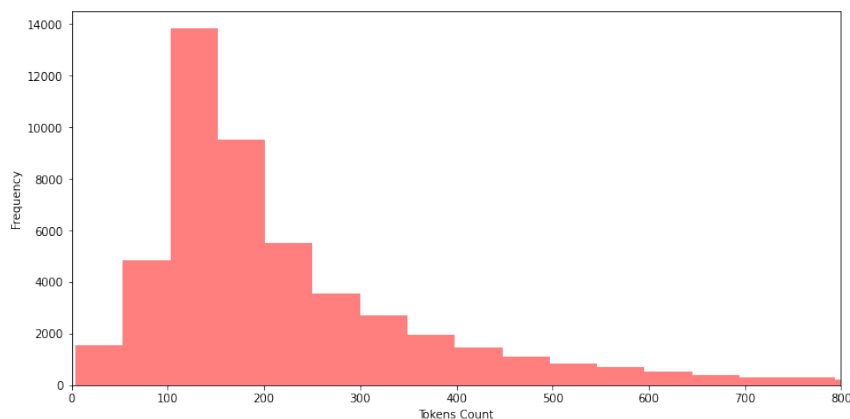


Figure 8. Distribution of the number of tokens for a better selection of the maximum sequence length

5.3. Effects of bias and training data

Carefully observing the accuracy and the loss learning curves in Figure 9 and Figure 10, we notice that the validation loss starts to creep upward and the validation accuracy starts to go down. In this perspective, the model in question continues to lose its ability to generalize well on unseen data. In fact, the model is relatively biased due to the effect of the training data and data-drift issues related to the fine-tuning data. In this context, we assume that the model starts to overfit. However, setting different dropouts, reducing the learning rate, or even trying larger batches will not work. On the other hand, these strategies sometimes give worst results, then a more critical overfitting problem. For this reason, pretraining these models on your industry data and vocabulary and then fine-tuning them may be the best solution.

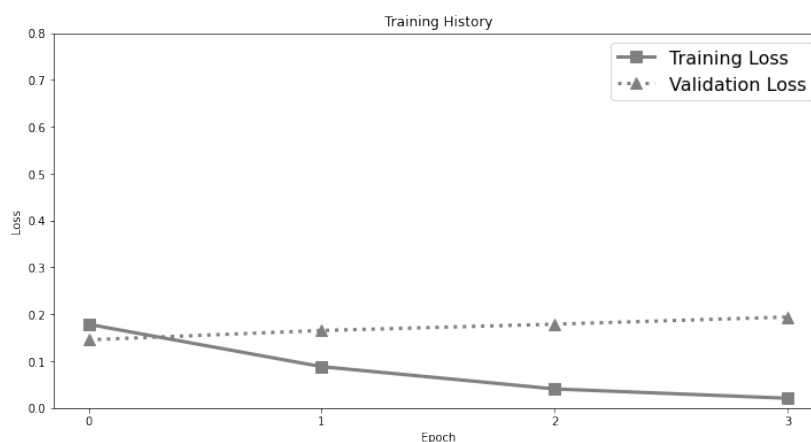


Figure 9. Best model: ELECTRA loss learning curve

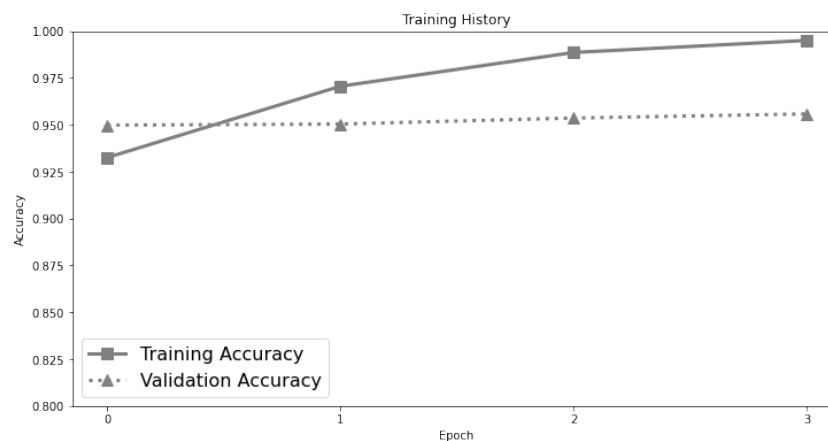


Figure 10. Best model: ELECTRA acc. learning curve

6. CONCLUSION

In this paper, we presented a detailed comparison to highlight the main characteristics of transformer-based pre-trained language models and what differentiates them from each other. Then, we studied their performance on the opinion mining task. Thereby, we deduce the power of fine-tuning and how it helps in leveraging the pre-trained models' knowledge to achieve high accuracy on downstream tasks, even with the bias they came with due to the pre-training data. Experimental results show how performant these models are. We have seen the highest F1-score with the ELECTRA model with 95.6 points, across the IMDB dataset. Similarly, we found that access to both left and right contexts is necessary when it comes to comprehension tasks like sentiment classification. We have seen that autoregressive models like GPT, GPT-2, and Reformer perform poorly and fail to achieve high accuracy. Nevertheless, XLNet has reached good results even though it is an autoregressive model because it incorporates ideas taken from encoders characterized by their bidirectional property. Indeed, all performances were nearby, including DistilBERT, which helps to gain incredible performance in less training time thanks to knowledge distillation. For example, for 4 epochs, BERT took 70 minutes to train, while DistilBERT took 35 minutes, losing only 0.6 F1 points, but saving half the time taken by BERT. Moreover, our ablation study shows that the maximum length of the sequence is one of the parameters having a significant impact on the final results and must be carefully analyzed and adjusted. Likewise, data quality is a must for good performance, data that will do not need to be processed, since extensive data cleaning processes may not help the model capture local and global contexts in sequences, distilled sometimes with words removed or trimmed during cleaning. Besides, we notice, that the majority of the models we fine-tuned on the IMDB dataset start to overfit at a certain number of epochs, which can lead to biased models. However, good quality data is not even enough, but pre-training a model on large amounts of business problem data and vocabulary may help on preventing it from making wrong predictions and may help on reaching a high level of generalization.

ACKNOWLEDGMENTS

We are grateful to the Hugging Face team for their role in democratizing state-of-the-art machine learning and natural language processing technology through open-source tools. Their commitment to providing valuable resources to the research community is highly appreciated, and we acknowledge their vital contribution to the development of our article.

APPENDIX

Appendix for "Analysis of the evolution of advanced transformer-based language models: experiments on opinion mining".

Table A1. Summary and comparison of transformer-based models

Model	L	H	A	Att. type	Total params	Tokenization	Training data	Computational cost	Training objectives	Performance tasks	Short description
GPT	12	512	12	Global	110M	Byte-pair-encoding [32]	Books Corpus (800M words)	-	Autoregressive, decoder	Zero-shot, text summarization, question answering, translation.	The first transformer-based autoregressive and causal masking model.
BERT	12	768	12	Global	110M	WordPiece [30]	Books Corpus (800M words) and English Wikipedia (2,500M words)	4 days on 4 Cloud TPUs in Pod configuration.	Autoencoding, encoder (MLM - NSP)	Text classification, natural language inference, question answering.	The first transformer-based autoencoding model, that uses global attention to provide high-level bidirectional contextualization.
GPT-2	12	1600	12	Global	117M	Byte-pair-encoding	WebText (10B words)	-	Autoregressive, decoder	Zero-shot, text summarization, question answering, translation.	Optimized and bigger than GPT and performs well on zero-shot settings.
GPT-3	96	12288	96	Global	175B	Byte-pair-encoding	Filtered Common Crawl, WebText2, Books1, Books2, and Wikipedia for 300B words.	-	Autoregressive, decoder	Text summarization, question answering, translation, zero-shot, one-shot, few-shot.	Bigger than its predecessors.
ALBERT	12	768	12	Global	11M	SentencePiece [31]	Books Corpus [35] and English Wikipedia.	Cloud TPU V3 TPUs number ranges from 64 to 512 (32h ALBERT-xxlarge).	Autoencoding, encoder, sentence-ordering prediction (SOP)	Semantic similarity, semantic relevance, question answering, reading comprehension.	Smaller and similar to BERT with minimal tweaks including the splitting of layers into groups via cross-layer parameter sharing, making it faster and reducing memory footprint.
DistilBERT	6	768	12	Global	66M	WordPiece	English Wikipedia [35] and Toronto Book Corpus.	90 hours on 8 16GB V100 GPUs.	Autoencoding (MLM), encoder	Semantic similarity, semantic relevance, question answering, textual entailment.	Pre-training leveraging knowledge distillation to deliver great results as BERT with lower latency.
RoBERTa	12	1024	12	Global	125M	Byte-pair-encoding	Book Corpus [35], CC-News, Open Web Text, and Stories [36].	8 32GB Nvidia V100 GPUs.	Autoencoding (Dynamic MLM, No NSP), encoder	Text classification, language inference, question answering.	Similar to BERT model but smaller. Pre-trained with large batches using some tricks for diverse learning like dynamic masking, where tokens are differently masked for each epoch.

XLM	12	2048	8	Global	-	Byte-pair encoding	Wikipedias of the XNLI languages.	64 Volta GPUs for the language modeling tasks and 8 GPUs for the MT tasks.	Autoencoding, encoder, causal language modeling (CLM), masked language modeling (MLM), and translation language modeling (TLM).	Translation tasks and NLU cross-lingual benchmarks.	By being trained on several pre-training objectives on a multilingual corpus, XLM proves that multilingual pre-training methods have a strong impact, especially on the performance of multilingual tasks.
XLM-RoBERTa	12	768	8	Global	270M	SentencePiece	CommonCrawl Corpus in 100 languages.	100 Nvidia V100 GPUs, 32GB	Autoencoding, encoder, MLM.	Translation tasks and NLU cross-lingual benchmarks.	Using only the masked language modeling objective, XLM-RoBERTa uses RoBERTa tricks on XLM approaches. It is able to detect the input language by itself (100 languages). Replaced token detection is a pre-training objective that addresses MLM issues and it results in efficient performance.
ELECTRA	12	768	12	Global	110M	WordPiece	Wikipedia, BooksCorpus, Gigas5 [37], ClueWeb 2012-B, and Common Crawl.	4 days on 1 GPU.	Generator (autoregressive, replaced token detection) and discriminator (Electra: predicting masked tokens).	Sentiment analysis, language inference tasks.	
DeBERTa	12	768	12	Global (Disentangled attention)	125M	Byte-pair encoding	Wikipedia, BooksCorpus, Reddit content, Stories, STORIES.	10 days 64 V100 GPUs.	Autoencoding, disentangled attention mechanism, and enhanced mask decoder.	DeBERTa was the first pretrained model to beat HLP on the SuperGLUE benchmark [38].	DeBERTa uses RoBERTa with disentangled attention and an enhanced mask decoder to significantly improve model performance on many downstream tasks while being trained only on half of the data used in RoBERTa large version.
XLNet	12	768	12	Global	110M	SentencePiece	Wikipedia, BooksCorpus, Gigas5 [37], ClueWeb 2012-B, and Common Crawl.	5.5 days on 512 TPU v3 chips.	Autoregressive, decoder	XLNet achieved state-of-the-art results and outperformed BERT on 20 downstream tasks including sentiment analysis, question answering, reading comprehension, document ranking.	XLNet incorporates ideas from transformer-XL [17] and addresses the pretraining finetune BERT's discrepancy being more capable to grasp dependencies between masked tokens.

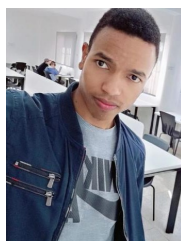
BART	12	768	16	Global	139M	Byte-pair encoding	Wikipedia, BooksCorpus.	-	Generative sequence to sequence, encoder-decoder, token masking, text infilling, sentence permutation, and document rotation.	BART beats its predecessors on generation tasks such as translation and achieved state-of-the-art results, while performing similarly to RoBERTa on discriminative tasks including question answering and classification.	Trained to map corrupted text to the original using an arbitrary noising function.
ConvBERT	12	768	12	Global	124M	WordPiece	OpenWebText [39]	GPU and TPU	Autoencoding, encoder	With fewer parameters and lower costs ConvBERT consistently outperforms BERT on various downstream tasks with less training cost.	For reduced redundancy and better modeling of global and local context, BERT's self-attention blocks are replaced by mixed-attention blocks incorporating self-attention and span-based dynamic convolutions.
Reformer	12	1024	8	Attention with local sensitive hashing	149M	SentencePiece	OpenWebText [39]	Parallelization across 8 GPUs or 8 TPU v3 cores.	Autoregressive, decoder.	Performs well with pragmatic requirements, thanks to reduction of the attention complexity.	An efficient and faster transformer that costs less time on long sequences thanks to two optimization techniques, local-sensitive hashing attention and axial position encoding.
T5	12	768	12	Global	220M	SentencePiece	The Colossal Clean Crawled Corpus (C4)	Cloud Pods.	Generative sequence to sequence, encoder-decoder.	Entailment, coreference challenges, question answering tasks via SuperGLUE benchmark	To incorporate the varieties of most linguistic tasks, T5 pre-trained on a mix of supervised and unsupervised tasks in a text-to-text format.
Longformer	12	768	12	Local + Global.	149M	Byte-pair-encoding	Books corpus, English Wikipedia, and Realnews dataset [40]	.	Autoregressive, decoder	Longformer achieved state-of-the-art results on two benchmark datasets WikiHop and TriviaQA.	For higher training efficiency on long documents, Longformer uses sparse matrices instead of attention matrices to linearly scale with sequences of length up to 4 096.




REFERENCES

- [1] K. R. Chowdhary, "Natural language processing," in *Fundamentals of artificial intelligence*, 2020, pp. 603–649, doi: 10.1007/978-81-322-3972-7_19.
- [2] M. Rhanoui, M. Mikram, S. Yousfi, A. Kasmi, and N. Zoubeidi, "A hybrid recommender system for patron driven library acquisition and weeding," in *Journal of King Saud University-Computer and Information Sciences*, 2020, vol. 34, no. 6, Part A, pp. 2809–2819, doi: 10.1016/j.jksuci.2020.10.017.
- [3] F. Z. Trabelsi, A. Khtira, and B. El Asri, "Hybrid recommendation systems: a state of art.," in *Proceedings of the 16th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE)*, 2021, pp. 281–288, doi: 10.5220/0010452202810288.
- [4] B. Pandey, D. K. Pandey, B. P. Mishra, and W. Rhmann, "A comprehensive survey of deep learning in the field of medical imaging and medical natural language processing: challenges and research directions," in *Journal of King Saud University-Computer and Information Sciences*, 2021, vol. 34, no. 8, Part A, pp. 5083–5099, doi: 10.1016/j.jksuci.2021.01.007.
- [5] A. Harnoune, M. Rhanoui, M. Mikram, S. Yousfi, Z. Elkaimbillah, and B. El Asri, "BERT based clinical knowledge extraction for biomedical knowledge graph construction and analysis," in *Computer Methods and Programs in Biomedicine Update*, 2021, vol. 1, p. 100042, doi: 10.1016/j.cmpbup.2021.100042.
- [6] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: a survey," in *Ain Shams engineering journal*, 2014, vol. 5, no. 4, pp. 1093–1113, doi: 10.1016/j.asej.2014.04.011.
- [7] S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," in *Information fusion*, 2017, vol. 36, pp. 10–25, doi: 10.1016/j.inffus.2016.10.004.
- [8] S. Yousfi, M. Rhanoui, and D. Chiadmi, "Mixed-profiling recommender systems for big data environment," in *First International Conference on Real Time Intelligent Systems*, 2017, pp. 79–89, doi: 10.1007/978-3-319-91337-7_8.
- [9] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 2019, vol. 1, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
- [10] A. Vaswani *et al.*, "Attention is all you need," in *Proceedings of the 31st Conference on Neural Information Processing Systems*, Dec. 2017, pp. 5998–6008, doi: 10.48550/arXiv.1706.03762.
- [11] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding with unsupervised learning," *Proceedings of the 2018 Conference on Neural Information Processing Systems*, 2018.
- [12] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
- [13] T. B. Brown *et al.*, "Language models are few-shot learners," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, vol. 33, pp. 1877–1901, doi: 10.48550/arXiv.2005.14165.
- [14] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *International Conference on Learning Representations*, 2019, doi: 10.48550/arXiv.1909.11942.
- [15] Y. Liu *et al.*, "RoBERTa: a robustly optimized BERT pretraining approach," *arXiv:1907.11692*, 2019, doi: 10.48550/arXiv.1907.11692.
- [16] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V Le, "XINet: generalized autoregressive pre-training for language understanding," in *Advances in neural information processing systems*, 2019, pp. 5753–5763, doi: 10.48550/arXiv.1906.08237.
- [17] Z. Yang, Y. Yang, J. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-XL: attentive language models beyond a fixed-length context," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Jul. 2019, pp. 2978–2988, doi: 10.18653/v1/P19-1285.
- [18] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019, doi: 10.48550/arXiv.1910.01108.
- [19] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, Mar. 2015, doi: 10.48550/arXiv.1503.02531.
- [20] G. Lample and A. Conneau, "Cross-lingual language model pretraining," *arXiv:1901.07291*, 2019, doi: 10.48550/arXiv.1901.07291.
- [21] A. Conneau *et al.*, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8440–8451, doi: 10.18653/v1/2020.acl-main.747.
- [22] M. Lewis *et al.*, "Bart: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880, doi: 10.18653/v1/2020.acl-main.703..
- [23] Z. Jiang, W. Yu, D. Zhou, Y. Chen, J. Feng, and S. Yan, "ConvBERT: Improving BERT with span-based dynamic convolution," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, p. 12, doi: 10.48550/arXiv.2008.02496.
- [24] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: the efficient transformer," *arXiv:2001.04451*, 2020, doi: 10.48550/arXiv.2001.04451.
- [25] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," in *Journal of Machine Learning Research*, 2020, vol. 21, no. 140, pp. 1–67, doi: 10.48550/arXiv.1910.10683.
- [26] K. Clark, M.-T. Luong, Q. V Le, and C. D. Manning, "Electra: pre-training text encoders as discriminators rather than generators," *arXiv:2003.10555*, p. 18, 2020, doi: 10.48550/arXiv.2003.10555.
- [27] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: the long-document transformer," *arXiv:2004.05150*, 2020, doi: 10.48550/arXiv.2004.05150.
- [28] P. He, X. Liu, J. Gao, and W. Chen, "DeBERTa: decoding-enhanced BERT with disentangled attention," *arXiv:2006.03654*, 2020, doi: 10.48550/arXiv.2006.03654.
- [29] S. Singh and A. Mahmood, "The NLP cookbook: modern recipes for transformer based deep learning architectures," in *IEEE Access*, 2021, vol. 9, pp. 68675–68702, doi: 10.1109/access.2021.3077350.




- [30] Y. Wu *et al.*, “Google’s neural machine translation system: Bridging the gap between human and machine translation,” in *arXiv preprint arXiv:1609.08144*, 2016, doi: 10.48550/arXiv.1609.08144.
- [31] T. Kudo and J. Richardson, “Sentence Piece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 66–71, doi: 10.18653/v1/D18-2012.
- [32] R. Sennrich, B. Haddow, and A. Birch, “Neural machine translation of rare words with subword units,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, vol. 1, pp. 1715–1725, doi: 10.18653/v1/P16-1162.
- [33] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, “Learning word vectors for sentiment analysis,” in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2011, vol. 1, pp. 142–150.
- [34] N. E. Zekaoui, “Opinion transformers.” 2023, [Online]. Available: <https://github.com/zekaouinouredine/Opinion-Transformers> (Accessed Jan. 2, 2023).
- [35] Y. Zhu *et al.*, “Aligning books and movies: towards story-like visual explanations by watching movies and reading books,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, vol. 2015 Inter, pp. 19–27, doi: 10.1109/ICCV.2015.11.
- [36] T. H. Trinh and Q. V. Le, “A simple method for commonsense reasoning,” in *arXiv:1806.02847*, 2018, doi: 10.48550/arXiv.1806.02847.
- [37] R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda, “English gigaword fifth edition, linguistic data consortium,” 2011, doi: 10.35111/wk4f-qt80.
- [38] A. Wang *et al.*, “SuperGLUE: A stickier benchmark for general-purpose language understanding systems,” in *Advances in neural information processing systems*, 2019, vol. 32, doi: 10.48550/arXiv.1905.00537.
- [39] A. Gokaslan and V. Cohen, “OpenWebText Corpus,” 2019. <http://skylion007.github.io/OpenWebTextCorpus> (Accessed Jan. 2, 2023).
- [40] R. Zellers *et al.*, “Defending against neural fake news,” *Advances in Neural Information Processing Systems*, vol. 32, p. 12, 2019, doi: 10.48550/arXiv.1905.12616.

BIOGRAPHIES OF AUTHORS






Nour Eddine Zekaoui    holds an Engineering degree in Knowledge and Data Engineering from School of Information Sciences, Morocco in 2021. He is currently a Machine Learning Engineer in a tech company. His research focuses on the areas of natural language processing and artificial intelligence, including information retrieval, question answering, semantic similarity, and bioinformatics. He can be contacted at email: nouredinezekaoui@gmail.com or nour-eddine.zekaoui@esi.ac.ma.






Siham Yousfi    is a Professor of Computer Sciences and Big Data at the School of Information Sciences, Rabat since 2011. She is a PhD holder from Mohammadia School of engineering of Mohammed V University in Rabat (2019). Her research interests include big data, natural language processing and artificial intelligence. She can be contacted at email: syousfi@esi.ac.ma.



Maryem Rhanoui    is an Associate Professor of Computer Sciences and Data Engineering at the School of Information Sciences, Rabat. She received an engineering degree in computer science then a PhD degree from ENSIAS, Mohammed V University, Rabat (2015). Her research interests include pattern recognition, computer vision, cybersecurity and medical data analysis. She can be contacted at email: mrhanoui@esi.ac.ma.



Mounia Mikram    is an Associate Professor of Computer Sciences and Mathematics at the School of Information Sciences, Rabat since 2010. She received her master degree from Mohammed V University Rabat (2003) and her PhD degree from Mohammed V University, Rabat, and Bordeaux I University (2008). Her research interests include pattern recognition, computer vision, biometrics security systems and artificial intelligence. She can be contacted at email: mmikram@esi.ac.ma.