An Evolutionary Overview of Large Language Models: From Statistical Methods to the Transformer Era

Boris Damjanović [1], Dragan Korać [2], Dejan Simić[3], Negovan Stamenković [4]

[1]Pan-European University Apeiron , Banja Luka, Bosnia and Herzegovina, boris.s.damjanovic@apeiron-edu.eu, https://orcid.org/0000-0003-4774-5774

[2]Faculty of Natural Sciences and Mathematics, University of Banja Luka, Banja Luka, Bosnia and Herzegovina, dragan.korac@pmf.unibl.org, 0000-0001-7798-5950

[3]Faculty of organizational sciences, University of Belgrade, Belgrade, Serbia, dejan.simic@fon.bg.ac.rs, https://orcid.org/0000-0002-0744-5411

[4]Pan-European University Apeiron, Banja Luka, Bosnia and Herzegovina, negovan.m.stamenkovic@apeiron-edu.eu, https://orcid.org/0000-0003-4025-5342

Preliminary communication

https://doi.org/10.7251/JIT2502145D UDC: 811.163.41`282.4:004.37

Abstract: While the early evolution of large language models (LLMs), including shift from statistical approaches to the Transformer architecture, illustrates their historical impact on the processing of natural language; however, the latest research in neural networks has enabled the faster and more powerful rise of language models grounded in solid theoretical foundations. These advantages, driven by advances in computing systems (e.g., ultra-powerful processing and memory capabilities), enable the development of numerous new models based on new emerging technologies such as artificial intelligence (AI). Thus, we provide an evolutionary overview of LLMs involved in the shift from the statistical to deep learning approach, highlighting their key stages of development, with a particular focused on concepts such as self-attention, the Transformer architecture, BERT, GPT, DeepSeek, and Claude. Finally, our conclusions present a reference point for future research associated with the emergence of new AI-supported models that are irreversibly transforming the way an increasing number of human activities are performed.

Keywords: Artificial intelligence, large language models, Transformer architecture, self-attention

Introduction

From early statistical approaches, such as Markov's contibution in 1913 to the Transformer architecture of 2017, large language models have greatly changed natural language processing . After the studies of Markov, the contributions of Shannon, who continued his research into language generation, and Jelinek, who investigated speech recognition, stand out. In 1990, Elman used a neural network called a multilayer perceptron to create simple recurrent networks with memory capable of processing simple languages.

Scientific studies on recurrent neural networks (RNN) from 1990 and convolutional networks (CNN) [6] from 1998 laid solid theoretical foundations for further research. A very important step in the development of LLM was the idea presented in the paper A Neural Probabilistic Language Model , in which the authors assigned a unique vector to each word, which they called embedding.

An important factor in the development of large language models was the emergence of increasingly powerful computer systems capable of processing and storage capabilities of large amounts of data. Parallel to this process, larger and larger data sets began to appear on which it was possible to train these models. In the development of large language models, the work Attention Is All You Need from 2017 stands out, in which the concept of self-attention and the Transformer architecture are presented, after which models such as BERT from 2018, GPT from 2018, and then many others emerged.

Today, due to the enormous progress of artificial intelligence and the appearance of a large number

of models, they are changing the way an increasing number of human activities are performed. This paper will present an overview of the history and key moments in the development of LLM with the most important scientific works that marked this field.

Early foundations and statistical language models

The development of large language models begins in the middle of the 20th century, when researchers used, by today's standards, small amounts of data. At that time, simple statistical techniques and relatively simple algorithms were used to predict the next word or phrase. Among the earliest studies that were the forerunners of today's models, Markov's study stands out . The research deals with the first 20,000 letters of Eugene Onegin's book. The letters were divided into vowels and consonants, then transitions between adjacent letters such as vowel-consonant or vowel-vowel were counted, and then probabilities were assigned to those transitions. Thus, Markov showed that the letters in the text are not independent and provided the first example of a Markov chain of order 1.

In his paper from 1948, Shannon models the language with a series of approximations. In the zero-order approximation, each letter is chosen independently and with the same probability, resulting in noise. In the first-order approximation, the letters are chosen according to their frequencies, which makes the text look a little more realistic. In the second-order approximation, which is called a bigram, the next letter depends on the previous one, so the generated string of letters already resembles human text. This approximation is also called the Markov chain of order 1. The approximation of the third order (trigram or Markov chain of order 2) is formed by the fact that the letter depends on the previous two. Now as a result we get recognizable word fragments. These simple models represented the initial steps in computational language generation. When it comes to text generation, the research of Jelinek , which was carried out at the IBM research center T.J. Watson Research Center in New York, is interesting. The primary focus of his research was speech recognition. He used statistics to create a model he called the IBM Raleigh language, which had 250 words. Using the Bayesian method, this model was able to generate sentences like "Each town is often without those services".

During the 1980s and 1990s, more advanced machine learning techniques such as support vector machines (SVMs), decision trees, naive Bayes and others emerged and were used for text classification. New discoveries in the field of neural networks enabled deeper progress in natural language understanding.

Successful application of neuronal approaches

In his paper from 1990 entitled *Finding Structure in Time* , Elman presents a simple recurrent network as a multilayer perceptron to which he added a unit that carries a copy of the previously hidden state. In this way, he created networks with memory that are able to process simple languages that resemble finite state machines. During learning, the network observes the current state t and the hidden state t-1. In this way, it creates separate clusters in which related terms, such as verbs and nouns, can be grouped together. Recurrent neural networks learn based on the error signal that is, based on the loss gradient when we go back through time. In the paper , the authors presented the application of the technique called *Gradient Based Learning* on convolutional networks. The gradient shows us where the algorithm went wrong in movement from the start to the desired goal. The error signal points in which direction and by how much weights should be moved in order to reach the goal as soon as possible. The problem that occurs is forgetting which occurs if the return error signal is multiplied many times by a number less than 1, or explosion if the error signal is multiplied by a number greater than 1. The problem of vanishing and exploding gradients was addressed by Hochreiter and Schmidhuber in their paper . They presented the *Long Short-Term Memory* cell, that is, a small circuit with memory and input, output and forget gates. This cell transmits information stably through many steps.

Vector representations of words (word embeddings)

A very important contribution in this area was *A Neural Probabilistic Language Model* , in which the authors presented a language model that assigns each word its unique vector representation, which they called embedding. In this way, they managed to achieve that similar words get vectors that are close. In the proposed approach, by observing the closeness

of the vectors, it is easy to generalize sentences like "The cat is walking in the bedroom", "A dog was running in a room", "The cat is running in a room" and many other combinations.

In the paper *Efficient Estimation of Word Representations in Vector Space*, the authors proposed a method to create word vectors, whose dimensions are typically from 100 to 300, in such a way that words appearing in similar contexts receive similar vectors. Because of this, relationships between words are obtained that allow conclusions such as the following: king – man + woman ≈ queen. The two basic components of this architecture are the *Continuous Bag-of-Words Model* and the *Continuous Skip-gram Model*. The CBOW model learns in such a way that it masks the middle word in the sentence, so it tries to guess from the context which word is missing. If the model made a mistake, it slightly adjusts the vectors to make it more accurate next time. *The Continuous Skip-gram Model* is another architecture that is similar to the CBOW architecture, but it takes the middle word and based on it predicts neighboring words in a smaller window around it. The paper provides an implementation in the programming language C, which is called word2vec. Unlike word2vec, in the paper *GloVe: Global Vectors for Word Representation* describing the GloVe architecture , the authors use the idea of expanding the window from which the model learns to the entire dataset. GloVe first counts how often each word appears next to every other, and then trains vectors on such data.

In the paper *Sequence to Sequence Learning with Neural Networks* , the authors demonstrated how *Long Short-Term Memory* networks can be used to encode the entire sentence into one vector of fixed length, for the purpose of translating text from English to French. This paved the way for further significant progress in natural language processing.

The attention mechanism and the transformer architecture

Before the publication of *Neural Machine Translation by Jointly Learning to Align and Translate* paper , neural machine translation was performed by creating one vector of fixed length based on the entire sentence. The attention mechanism presented by the authors assigns different weights to all parts of the input sentence such that weights sum to 1. This attention mechanism is also called soft alignment because it distributes attention to several words instead of one choice. For example, if it needs to translate the word bank, attention will give more weight to input words like credit and money, and less weight to words like dog and cat. The concepts presented in this paper have significantly improved machine translation and laid the foundation for subsequent research in this area.

In the paper *Effective Approaches to Attention-based Neural Machine Translation* the authors propose two attention mechanisms for neural machine translation – a global approach in which all positions of input tokens are always observed and a local approach in which only a subset of sequence positions is observed. Following the publication of the previously described papers, a larger number of contributions in different fields appear. This is how the paper *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation* that deals with machine translation, *A neural attention model for abstractive sentence summarization* that deals with text summarization, and *Listen, attend and spell* dedicated to speech recognition and *Show, attend and tell: Neural image caption generation with visual attention* to generate image captions appears.

Google's *Neural Machine Translation* (GNMT) was the first neural translator widely used in practice. It consisted of a deep LSTM network with eight encoder and eight decoder layers. In order to improve parallelism, the models attention mechanism connected bottom layer of the decoder with the top layer of the encoder. To improve the translation of rare words, this model divided the words into a limited, pre-fixed, set of common word pieces known as *„wordpieces"*.

A very important study that laid the foundations for further scientific research is *Attention Is All You Need* , in which the authors use a self-attention mechanism that allows the model to learn which words are related to each other. They also introduce a new concept called multi-head attention which uses several attention heads that capture different types of relationships, that is, the use of several "heads" that simultaneously explore different types of relationships between words.

Unlike the previously described techniques, the self-attention mechanism connects distant words directly, without sequential processing. The authors

also introduce the Transformer architecture, which consists of an encoder and a decoder. The encoder's self-attention mechanism was unmasked, making it bidirectional. Each word is associated with all input words to the left and right of the observed word. The decoder uses a masked attention mechanism, in the sense that it is only allowed to "look" backwards. In the paper, the authors present another idea, they add positional encodings to the words, thus taking care of the order of the words. This model processes tokens concurrently, which makes it very suitable for GPU/TPU execution, which was later adopted by many other models.

BERT and bidirectional context understanding

A new stage in the development of large language models begins with the work of a group of researchers from Google in which the authors present a language representation model called BERT (*Bidirectional Encoder Representations from Transformers)*. As its name suggests, BERT was the first large and widely accepted bidirectional model that could observe an entire sentence. BERT is primarily intended as a powerful tool for in-depth understanding of language and context.

The model was pretrained using two techniques: *Masked Language Modeling (MLM)* and *Next Sentence Prediction (NSP*). The MLM training technique was performed in such a way that, in a random manner, 15% of the tokens in the sentences were hidden and the model was left to try to find the hidden (masked) words. The idea of the NSP technique was to give the model two sentences, and for the model to try to determine whether the second sentence really follows the first or if the second sentence is some random content.

In the fine tuning phase, a small task specific output layer is added on top of the pretrained model. When BERT receives some training text at the input, that text passes through the entire network that belongs to the model, in order to generate context vectors. At the output layer, based on them, a prediction is calculated, and then, based on the prediction and correct labels, the loss is calculated. Based on the loss, a gradient vector is obtained, which is propagated backwards through the entire model and fine-tunes the model so that the next time the error is smaller.

After the introduction of the BERT model, a range of models based on this architecture appeared. RoBERTa is a model that achieves better results thanks to a modified training regime. The DistilBERT model represents a distilled version of BERT that, thanks to a different learning method (knowledge distillation) in the pretraining phase, reduced the size of the model by 40%, while retaining 97% of language comprehension efficiency and being 60% faster. In the ALBERT model the authors focus on the problem of GPU memory and time consumption with model scaling, and present techniques for reducing the number of parameters with the same or better accuracy. The BERTić model [28] is a South Slavic version of BERT that was pretrained using 8 billion tokens from web pages that contained Bosnian, Croatian, Serbian and Montenegrin languages.

The GPT series: generative models

Unlike BERT, which is a bidirectional encoder, the authors of the paper *Improving Language Understanding by Generative Pre-Training* introduced a transformer-based language model pretrained using a generative objective that functions as a decoder that observes only the left context and tries to predict the right one. The authors have shown that such a model can be used for various tasks, such as text implication, answering questions, evaluating semantic similarity, as well as document classification. It should be noted that this model, due to the fact that it used the Generative Pre-Training procedure, was later named the Generative Pre-Training Transformer or GPT-1.

Unlike earlier models that were trained on datasets prepared for supervised learning, the authors of *Language Models are Unsupervised Multitask Learners* demonstrate the fact that really large models can learn without supervision if they are trained on sufficiently large datasets. For training purposes, the authors created a corpus of data they named WebText, which consisted of millions of web pages. A model trained on this corpus, without any task-specific learning (zero-shot), achieved the best results on 7 out of 8 language modeling benchmark datasets. The authors state that using this model they were able to obtain very coherent paragraphs of text.

In the paper *Universal Language Model Fine-tuning for Text Classification* , the authors present *ULMFiT* model. This model is first trained on the large

corpus of WikiText-103, and then transfer of knowledge is carried out, i.e., fine-tuning for the target task (*transfer learning*).

By creating a language model of 175 billion parameters, the authors of the paper *Language Models are Few-Shot Learners* demonstrated that the model named GPT-3 can learn a new task from just a few examples in a query (few-shot learning) without additional training. They used Filtered Common Crawl, WebText2, two Book corpora, and Wikipedia as their training datasets. Their idea was that without fine-tuning, the model could improve on various tasks, from translation, question answering to reasoning, simply by increasing the training dataset and model size. This model could write persuasive texts, discuss and answer questions on various topics, generate computer code, and solve simple math problems.

A large number of studies have shown that scaling language models predictably improves their performance. In the paper *Emergent Abilities of Large Language Models* , a different and less predictable property called the emergent ability of large language models is discussed. An ability is considered emergent if it does not exist in smaller models and appears in larger ones. The paper examines a large number of models that are unable to solve problems with small training datasets, and then their performance suddenly improves after crossing a certain threshold.

In the paper *Training language models to follow instructions with human feedback* , the authors point out the fact that the best results are not necessarily achieved simply by enlarging large language models. They trained the model using a fine-tuning process in which humans (raters) write questions and good answers. Then, for the same question, the model offers more answers, and people choose the better one, thus fine-tuning the models. Such models are referred to as *instructGPT*. In the OpenAI blog titled Introducing ChatGPT it is stated that the very concept of *instructGPT* was used to create ChatGPT which was obtained by fine-tuning the GPT-3.5 model.

The report called GPT-4 Technical Report states that GPT-4 is a large model that can take text and images at the input, and producing text at the output. The team that worked on its development, due to competition and security concerns, did not give much information about its architecture. It was designed as a multimodal Transformer that in many fields, both professional and academic, has shown results that can be compared to human ones. However, he still makes mistakes, hallucinates, and shows bias on a whole range of topics. The GPT-4.5 model was presented at the end of February 2025 in a document called OpenAI GPT-4.5 System Card . During the creation of this, until then, their largest model, some traditional techniques (*reinforcement learning from human feedback and supervised fine-tuning*) and some new supervised learning techniques were used, which are not explained in the document. With this model, improvements were achieved in emotional intelligence, the ability to write, program and practical problems solving. The document states that this was a research preview version of the model, emphasing that its capabilities were still being explored at that time.

At the moment, the latest model, GPT-5, was introduced on August 7, 2025 in a document titled GPT-5 System Card . GPT-5 is not a single model, but a unified system consisting of several models and a router that selects the appropriate model depending on the question. The system consists of gpt-5-main, gpt-5-main-mini, gpt-5-thinking, gpt-5-thinking-mini, gpt-5-thinking-nano, and gpt-5-thinking-pro.

The first two models in the list (gpt-5-main, gpt-5-main-mini) are high throughput, while the gpt-5-thinking model consumes the most computing power for reasoning, and gpt-5-thinking-pro uses parallelization in reasoning to get the best answer.

Other notable models

While OpenAI continued to develop the GPT family of models, other research groups were working on their own approaches. In the paper *LLaMA: Open and Efficient Foundation Language Models* the authors describe the LLaMA model, whose main idea is that greater efficiency can be achieved by using much larger datasets for training while creating smaller models. The authors claimed that LLaMA-13B achieves better results than GPT-3 with 175B parameters, while LLaMA-65B competed with the then best models at the time. The next version, Llama 2 described in paper *Llama 2: Open Foundation and Fine-Tuned Chat Models* , was released as a collection of large language models that includes models from 7 to 70 billion parameters. In the paper *The Llama 3 Herd of Models* the authors present Llama 3.1 as a family of models

supporting multilingualism, coding, reasoning with the largest model having 405 billion parameters. The article *The future of AI: Built with Llama* states that after the first Meta multimodal model Llama 3.2, Llama 3.3 was created as a text model that gives the same results as Llama 3.1 405B, but with only a fraction of the resource consumption. In the paper *The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation* the two latest Llama 4 models - Scout and Maverick - were presented. Both are 17B parameters and are Mixture-of-Experts type, meaning that they use only a small part of the subnet (experts) for each question.

Papers describing DeepSeek LLM and DeepSeek-Coder appeared in January 2024. DeepSeek LLM is an open-source model trained from scratch on a massive dataset of approximately 2 trillion tokens. After that, the model was trained using supervised fine-tuning and direct preference optimization techniques (DPO). This led to the creation of the DeepSeek LLM 7B and DeepSeek LLM 67B chat models. The authors claim in the paper that DeepSeek LLM 67B demonstrated superior performance compared to LLaMA-2 70B and GPT-3.5. DeepSeek-Coder was developed to encourage research and development outside the circle of closed source models. This model was released in three versions 1.3B, 6.7B and 33B parameters. Then the authors present DeepSeek-Coder v1.5, which, in addition to coding tasks, also shows a good understanding of natural language. Improved versions appeared soon, such as DeepSeek-V2 which introduces Multi-head Latent Attention, a mechanism that reduce the key-value cache and DeepSeek-MoE technique which efficiently uses subnets experts. This model consisted of 236 billion total and 21 billion parameters per token and supported a context length of 128000 tokens. The DeepSeek-V3 model has 671 billion in total and 37 billion active parameters per token. In addition, it introduces a multi-token prediction technique in addition to the classic prediction of the next token to improve performance. In their paper the authors present the DeepSeek-R1-Zero and DeepSeek-R1 models. DeepSeek-R1-Zero shows that a model capable of strong reasoning can only be made through reinforcement learning. DeepSeek-R1 improves reasoning using multi-level learning and a technique called cold-start, in which model training begins with a small but well proven dataset, to get a more stable learning start. The authors have released as open source DeepSeek-R1-Zero and DeepSeek-R1 and 6 more models 1.5B, 7B, 8B, 14B, 32B, 70B, distilled from DeepSeek-R1, based on Qwen and Llama architectures.

In the report entitled *Gemini: A Family of Highly Capable Multimodal Models* , the authors present the family of Gemini models, which consists of the Ultra, Pro, and Nano model. These models could process image, sound, video and text, and output text. This was Google's first family of general-purpose models trained from the ground up to work with image, audio, video and text simultaneously. In March 2024, the Gemini 1.5 was introduced in paper *Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context* , consisting of an upgraded Gemini 1.5 Pro model and a slightly lighter Gemini 1.5 Flash variant. The announcement of Gemini 2.0 was followed by its presentation in an article *Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities* along with two new Gemini 2.5 models: Gemini 2.5 Pro and 2.5 Flash. Gemini 2.5 Pro is a model that excels in multimodal data processing and can process up to 3 hours of video.

The model called Mistral 7B with 7 billion parameters is the first model in this family that uses Grouped-Query Attention to increase the processing speed and Sliding Window Attention to enable handling of long sequences. The Mixtral 8x7B has the same architecture as the Mistral 7B, but in addition includes 8 "feed-forward" blocks (experts). For each token, the router chooses two experts to process it. Then the processing results are combined and a better result is obtained. The next product of this company was the Mistral Large , one of the large models that was available through the API. It should be noted that their portfolio includes a model called Codestral , which is designed for coding, having knowledge of 80 programming languages.

The paper *Constitutional AI: Harmlessness from AI Feedback* presented an idea for a new model of artificial intelligence that, instead of relying on human knowledge and work to improve itself, teaches itself how to be useful and safe. In that process, the model uses a "constitution", that is, a set of clear guidelines that lead it to be better through two steps. The model first gives a crude answer, then criticizes and refines

it with the help of the constitution. Then the answer is reinforced with the *Reinforcement Learning from AI Feedback*, in which the second model acting as a judge chooses the better one from the two offered answers. This idea inspired the authors of the Claude model series whose latest model is the Claude Opus 4.1 .

In November 2023, xAi introduced the Grok model which was soon released as an open source model with 314 billion parameters per model, which uses a multi-expert (Mixture of Experts) MoE architecture, where the router selects a smaller subset of experts for each problem. Other versions Grok-1.5 Vision, Grok-2, Grok-3, Grok-4 were only available as commercial models. Currently the most powerful model Grok-4 is only available in Grok app with SuperGrok or Premium+ account and via xAi API. It was trained using reinforcement learning on a Colossus cluster computer with 200,000 GPUs.

In addition to the general-purpose large models presented above, there are also models that have been created for specific tasks. We have already mentioned some of the models intended for coding (DeepSeek Coder, Codestral). This group also includes GitHub Copilot and Gemini Code Assist. Some of the models that are intended to solve mathematical problems are MathGPT, Minerva, WizardMath, MathGLM, Llemma . Among the biomedical models specialized in literature searching and question answering, we will point out BioGPT, GatorTron, MEDITRON, PMC-LLaMA, BioMegatron, Med-PaLM . There are a number of models that are intended to generate an image based on the description, such as Stable Diffusion, DALL-E 3, Midjourney, Imagen, etc. It should be taken into account that we have listed here only the most famous models and that the list does not end here, as well as that general purpose models can often be effectively applied to a wide range of problems . Also, when it comes to models that are created for special purposes, the list does not end here, because there are numerous models for different purposes that are not mentioned here - from health, economy, finance, transport and logistics, security, energy, agriculture, education, science and research, mathematics, robotics, multimedia, law and many other fields.

Conclusion

This overview, based on the development of large language models, presents their evolutionary flow observed from basic statistical to deep learning approches, with aim to identify two key pillars. First, it consists of used architectures and techniques such as RNN/LSTM, attention mechanism, MoE (Mixture of Experts), GNMT (Google Neural Machine Translation), or Transformer architecture, while second, it consists of training the model with an increase in the dataset size and techniques such as masked language modeling (MLM), next sentence prediction (NSP), supervised fine-tuning, direct preference optimization and reinforcement learning from AI feedback. Also, as our results shows currently advantages of closed models in terms of capabilities and performance; however the significance of open source-models provide tremendous opportunities for all future research effort with approximate performance. Finally, our conclusion indicates an increasing trend of development of general-purpose models, focusing on large language models and artificial intelligence that will influence all areas of life.

References

[1] А. А. Марков, "Пример статистического исследования над текстом «Евгения Онегина», иллюстрирующий связь испытаний в цепь.," *Известия Императорской Академии наук, серия VI, том VII, Санкт-Петербург, ,* p. стр. 153–162., 1913.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez and Ł. Kaiser, "Attention Is All You Need," *In Advances in Neural Information Processing Systems,* 2017.

[3] D. Jurafsky and J. H. Martin, Speech and Language Processing (2nd ed.), Pearson Education, 2009.

[4] K. R. Chowdhary, Natural Language Processing. Fundamentals of Artificial Intelligence, Springer, 2020.

[5] J. L. Elman, "Finding Structure in Time," *Cognitive Science,* 1990.

[6] Y. Bengio, R. Ducharme, P. Vincent and C. Jauvin, "A Neural Probabilistic Language Model," *Journal of machine learning research,* 2003.

[7] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv,* 2018.

[8] A. Radford, K. Narasimhan, T. Salimans and I. Sutskever, "Improving Language Understanding by Generative Pre-Training," *OpenAI preprint,* 2018.

[9] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal,* p. 379–423, 1948.

[10] F. Jelinek, "Continuous Speech Recognition by Statistical Methods," *Proceedings of the IEEE,* p. 532–556, 1976.

[11] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," *European conference on machine learning,* p. Springer, 1998.

[12] T. MITCHELL , T. M., Machine Learning, New York: McGraw Hill, 1996.

[13] D. LEWIS, "Naive (Bayes) at forty: The independence as-

sumption in information re-trieval," in *In Proceedings of ECML-98, 10th European Conference on Machine Learning*, Chemnitz, Germany, 1998.
[14] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE,* 1998.
[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation, ieeexplore.ieee.org,* 1997.
[16] J. Pennington, R. Socher and C. D. Manning, "GloVe: Global Vectors for Word Representation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* 2014.
[17] I. Sutskever, O. Vinyals and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," *Advances in neural information processing systems,* 2014.
[18] D. Bahdanau, K. H. Cho and Y. Bengio, Neural Machine Translation by Jointly Learning to Align and Translate, arXiv, 2014.
[19] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi and ..., "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation," *arXiv preprint,* 2016.
[20] A. M. Rush, S. Chopra and J. Weston, "A neural attention model for abstractive sentence summarization," *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP),* p. 379–389, 2015.
[21] W. Chan, N. Jaitly, Q. Le and O. Vinyals, "Listen, attend and spell," *A neural network for large vocabulary conversational speech recognition. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* p. 4960–4964, 2016.
[22] K. Xu, J. Lei Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015) ,* p. 2048–2057, 2015.
[23] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, 2019.
[24] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv:1907.11692,* 2019.
[25] V. Sanh, L. Debut, J. Chaumond and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv:1910.01108,* 2019.
[26] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," *arXiv:1909.11942,* 2019.
[27] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei and I. Sutskever, "Language Models are Unsupervised Multitask Learners," *OpenAI preprint,* 2019.
[28] J. Howard and S. Ruder, "Universal Language Model Fine-tuning for Text Classification," *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics,* p. 328–339, 2018.
[29] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child and Ram, "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems (NeurIPS),* 2020.
[30] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean and W. Fedus, "Emergent Abilities of Large Language Models," *arXiv:2206.07682,* 2022.
[31] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell and Welinder, "Training language models to follow instructions with human feedback," *arXiv:2203.02155,* 2022.
[32] OpenAI, "Introducing ChatGPT," *OpenAI,* November 30, 2022.
[33] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. Leoni Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom and P. Baltescu, "GPT-4 Technical Report," *OpenAI,* 2023.
[34] OpenAI, "OpenAI GPT-4.5 System Card," 2025.
[35] OpenAI, "GPT-5 System Card," *2025.*
[36] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave and G. Lample, "LLaMA: Open and Efficient Foundation Language Models," *arXiv:2302.13971 ,* 2023.
[37] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton Ferrer, M. Chen, G. Cucurull and Esiob, "Llama 2: Open Foundation and Fine-Tuned Chat Models," *arXiv:2307.09288,* 2023.
[38] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang and A. Mitra, "The Llama 3 Herd of Models," *arXiv:2407.21783,* 2024.
[39] Meta, "The future of AI: Built with Llama," *Meta,* 2024.
[40] Meta, "The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation," *Meta,* 2025.
[41] X. Bi, D. Chen, G. Chen, S. Chen, D. Dai, C. Deng, H. Ding, K. Dong, Q. Du, Z. Fu, H. Gao, K. Gao, W. Gao, R. Ge, K. Guan, D. Guo, J. Guo, G. Hao, Z. Hao, Y. He and H, "DeepSeek LLM, Scaling Open-Source Language Models with Longtermism," *arXiv:2401.02954,* 2024.
[42] D. Guo, Q. Zhu, D. Yang, Z. Xie, K. Dong, W. Zhang, G. Chen, X. Bi, Y. Wu, Y. Li, F. Luo, Y. Xiong and W. Liang, "DeepSeek-Coder: When the Large Language Model Meets Programming -- The Rise of Code Intelligence," *arXiv:2401.14196,* 2024.
[43] A. Liu, B. Feng, B. Wang, B. Wang, B. Liu, C. Zhao, C. Dengr, C. Ruan, D. Dai, D. Guo, D. Yang, D. Chen, D. Ji, E. Li, F. Lin, F. Luo, G. Hao, G. Chen, G. Li and H. Zhang, "DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model," *arXiv:2405.04434,* 2024.
[44] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, D. Dai, D. Guo, D. Yang, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, F. Luo and G. Hao, "DeepSeek-V3 Technical Report," *arXiv:2412.19437,* 2024.
[45] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, B. Xue and Wan, "DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning," *arXiv:2501.12948,* 2025.
[46] R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, D. Silver, M. Johnson, I. Antonoglou, J. Schrittwieser, A. Glaese, J. Chen and P, "Gemini: A Family of Highly Capable Multimodal Models," *arXiv:2312.11805 ,* 2023.

[47] P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang, S. Mariooryad, Y. Ding, X. Geng, F. Alcober, R. Frostig, M. Omernick, L. Walker and C. Paduraru, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," *arXiv:2403.05530,* 2024.

[48] G. GeminiTeam, "Gemini 2.0: Our latest, most capable AI model yet," *Google DeepMind,* 2024.

[49] G. GeminiTeam, "Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities.," *Google DeepMind,* 2025.

[50] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock and Le, "Mistral 7B," *arXiv:2310.06825,* 2023.

[51] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. Bou Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample and L. Lavaud, "Mixtral of Experts," *arXiv:2401.04088,* 2024.

[52] MistralAI, "Au Large," 2024. [Online]. Available: https://mistral.ai/news/mistral-large.

[53] Mistral, "Codestral 25.01," 2025. [Online]. Available: https://mistral.ai/news/codestral-2501.

[54] Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, C. Chen, C. Olsson, C. Olah, D. Hernandez, D. Drain and D. Ganguli, "Constitutional AI: Harmlessness from AI Feedback," *arXiv:2212.08073,* 2022.

[55] Anthropic, "The Claude 3 Model Family: Opus, Sonnet, Haiku," *Claude CDN,* 2025.

[56] Anthropic, "Claude Opus 4.1," 2025. [Online]. Available: https://www.anthropic.com/claude/opus.

[57] xAi, "Announcing Grok," 2023. [Online]. Available: https://x.ai/news/grok.

[58] xAi, "Open Release of Grok-1," xAi, 2024. [Online]. Available: https://x.ai/news/grok-os.

[59] xAi, "Grok 4," 2025. [Online]. Available: https://x.ai/news/grok-4.

[60] W. X. Zhao, K. Zhou and J. Li, "A Survey of Large Language Models," *arXiv:2303.18223,* 2023.

[61] D. Korać, B. Damjanović and D. Simić, "A model of digital identity for better information security in e-learning systems," *The Journal of Supercomputing ,* 2022.

[62] C. Pu, J. Seol, N. Park and D. Korac, "Authenticated Key Agreement Protocol for Device-to-Gateway Communication in IoT," in *IEEE Consumer Communications & Networking Conference (IEEE CCNC 2026)*, 2026.

[63] D. Korać, B. Damjanović, D. Simić and C. Pu, "Management of evaluation processes and creation of authentication metrics: Artificial intelligence-based fusion framework," *Information Processing & Management,* 2025.

[64] R. Bommasani, D. A. Hudson and E. Adeli, "On the Opportunities and Risks of Foundation Models," *Stanford CRFM Report,* 2021.

[65] D. Korać, D. Čvokić and D. Simić, "Computational Engineering Approach-Based Modeling of Safety and Security Boundaries: A Review, Novel Model, and Comparison," *Archives of Computational Methods in Engineering ,* 2025.

[66] D. Korać, B. Damjanović, D. Simić and K.-K. R. Choo, "A hybrid XSS attack (HYXSSA) based on fusion approach: Challenges, threats and implications in cybersecurity," *Journal of King Saud University - Computer and Information Sciences,* 2025.

[67] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed., Pearson, 2021.

[68] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.

Received: November 7, 2025
Accepted: November 12, 2025

About the authors

Boris Damjanović is an Associate Professor at the Faculty of Information Technology, Pan-European University Apeiron in Banja Luka. Contact: boris.s.damjanovic@apeiron-edu.eu Areas of interest: data protection in computer systems, artificial intelligence, programming languages, and the application of information technologies.

Dragan Korać is an Associate Professor in the Department of Computer Science, Faculty of Natural Sciences and Mathematics, University of Banja Luka, Bosnia and Herzegovina. He received his Ph.D. in Computer Science and Informatics in 2018 from the Faculty of Organizational Sciences, University of Belgrade. His main research interests include cybersecurity, information security, and mobile computing.

Negovan Stamenković received the B.Sc. degree in Electronics and Telecommunications from the Faculty of Technical Sciences, Kosovska Mitrovica, Serbia, in 2006, and the Ph.D. degree in Electronic Engineering from the Faculty of Electronic Engineering, University of Nis, Serbia, in 2011. He is currently a ˇ Full Professor at the Apeiron, University in BanjaLuka, BiH. His major research interests include digital signal processing, computer engineering, and modular arithmetic.

For citation

Boris Damjanović, Dragan Korać, Dejan Simić, Negovan Stamenković, An Evolutionary Overview of Large Language Models: From Statistical Methods to the Transformer Era, *JITA – Journal of Information Technology and Applications, Banja Luka*, Pan-Europien University APEIRON, Banja Luka, Republika Srpska, Bosna i Hercegovina, JITA 15(2025)2:145-153, (UDC: 811.163.41`282.4:004.37), (DOI: 10.7251/JIT2502145D), Volume 15, Number 2, Banja Luka, December (81-176), ISSN 2232-9625 (print), ISSN 2233-0194 (online), UDC 004