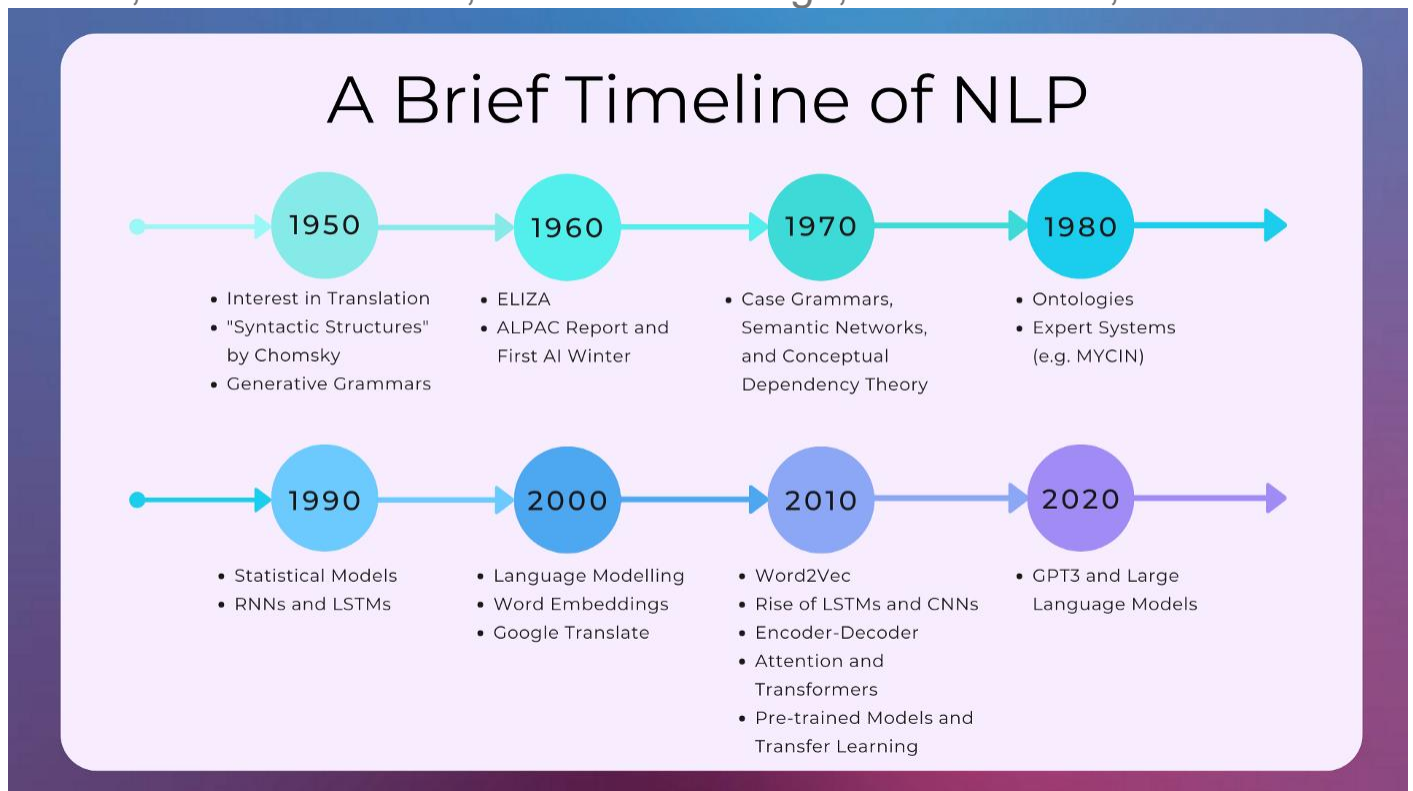


# A Brief Timeline of NLP

A journey across grammars, expert systems, ontologies, statistical models, neural networks, word embeddings, transformers, etc.



A brief timeline of NLP. Image by the author.

## What is NLP

Natural Language Processing (NLP) is a subfield of artificial intelligence that is concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze natural language data.

NLP is used to apply computational methods to text in order to extract meaning from it in a way that is similar to the way humans do. This can be used for a variety of tasks, such as automated machine translation, sentiment analysis, topic identification, and text summarization.

NLP is a very active area of research, with new techniques and applications being developed all the time. Thanks to the advent of machine learning and deep learning, the quality of NLP results has improved a lot, thus fueling a great variety of valuable applications.

To be clearer about what NLP is capable of doing today, here are a couple of examples that I find particularly significant.

### **Generating Draft Articles**

There are NLP models that can generate draft articles written with quality very similar to that of a human writer. We can ask one of these models to write a small article explaining why sleep is important.

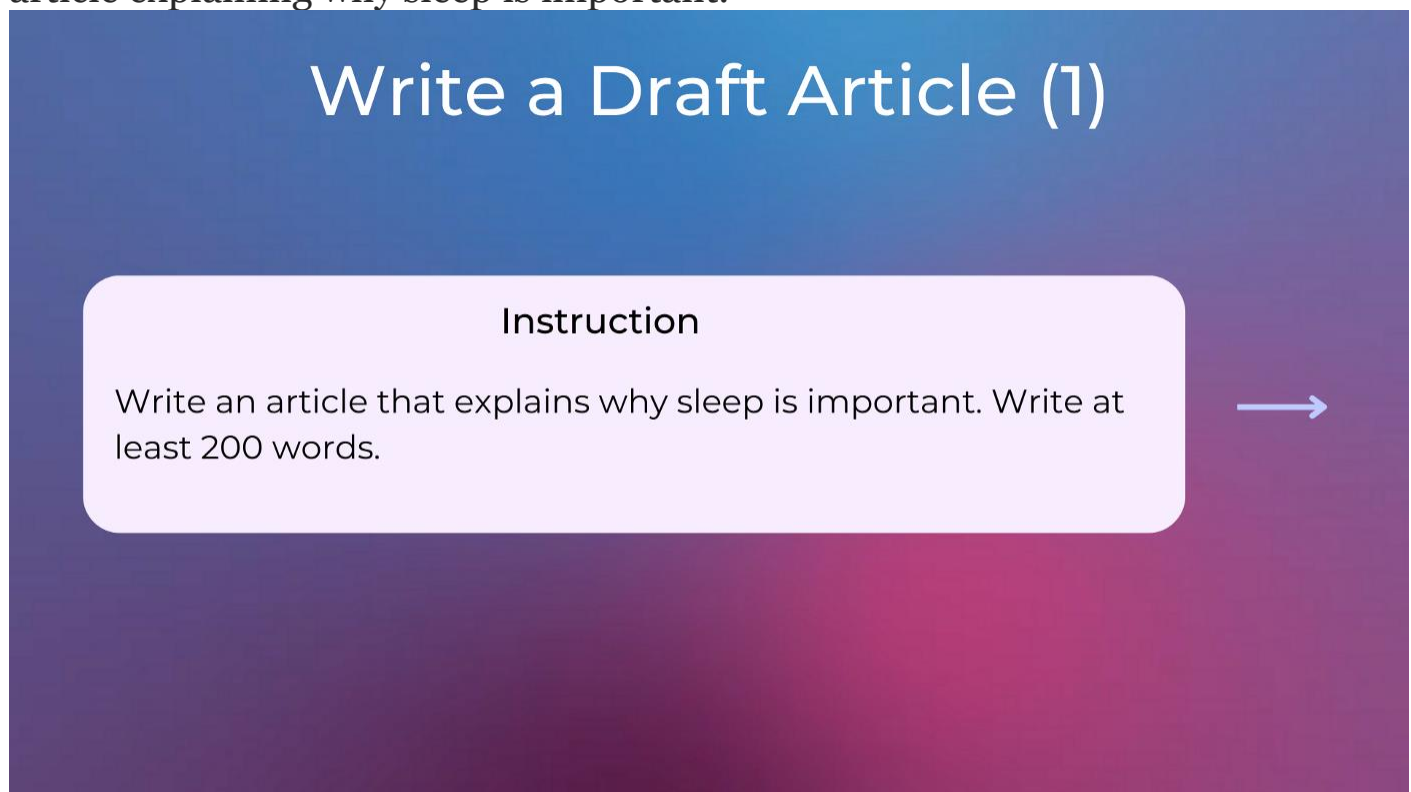


Image by the author.

Here follows the article generated by the model (not cherry-picked).

# Write a Draft Article (2)

## Generated Article

→ Sleep is critical for our physical, mental, and emotional health. It is the time when our bodies heal, repair damage, and grow. Without enough quality sleep, we are at increased risk for chronic diseases, such as heart disease, diabetes, and obesity. We are also more likely to experience anxiety, depression, and mood swings.

Sleep deprivation can lead to errors, accidents, and poor decision-making. It can also impact our memory, learning, and creativity. In fact, sleep deprivation is one of the main contributing factors to road accidents.

We need sleep to function at our best. When we don't get enough sleep, our bodies and minds pay the price. So, make sure to get enough quality sleep each and every night!

Image by the author.

As you can see, the generated article has very high quality and is very similar to what a writer would do. This is possible because this model has been trained on a huge amount of texts from around the Internet, trying to learn the distribution of words in the language. Therefore, when the model reads the instruction about writing an article about sleep, it generates the next most plausible text, rearranging the knowledge it has acquired during its training. The written article is therefore a reorganization of his acquired knowledge on training data, it is not a copy/paste from an already existing article.

## ***Language Models***

*A model like the one in the previous examples is called **language model**. A language model is trained on a huge amount of texts, trying to learn the distribution of the words in the language. This can be done by training the*

model to predict the words that follow a sentence, or trying to recover masked words from their surrounding context.

## **GPT3**

The previous examples have been produced using the [GPT3](#) language model. This model is a **transformer** neural network consisting of 175 billions of parameters and requiring approximately 800GB of storage. The training data consists of hundreds of billions of words from texts from around the Internet. It has been trained by OpenAI, which hosts it and make it available as a service. GPT3 and its experiments are described in the paper [Language Models are Few-Shot Learners](#).

## **The Turing Test**

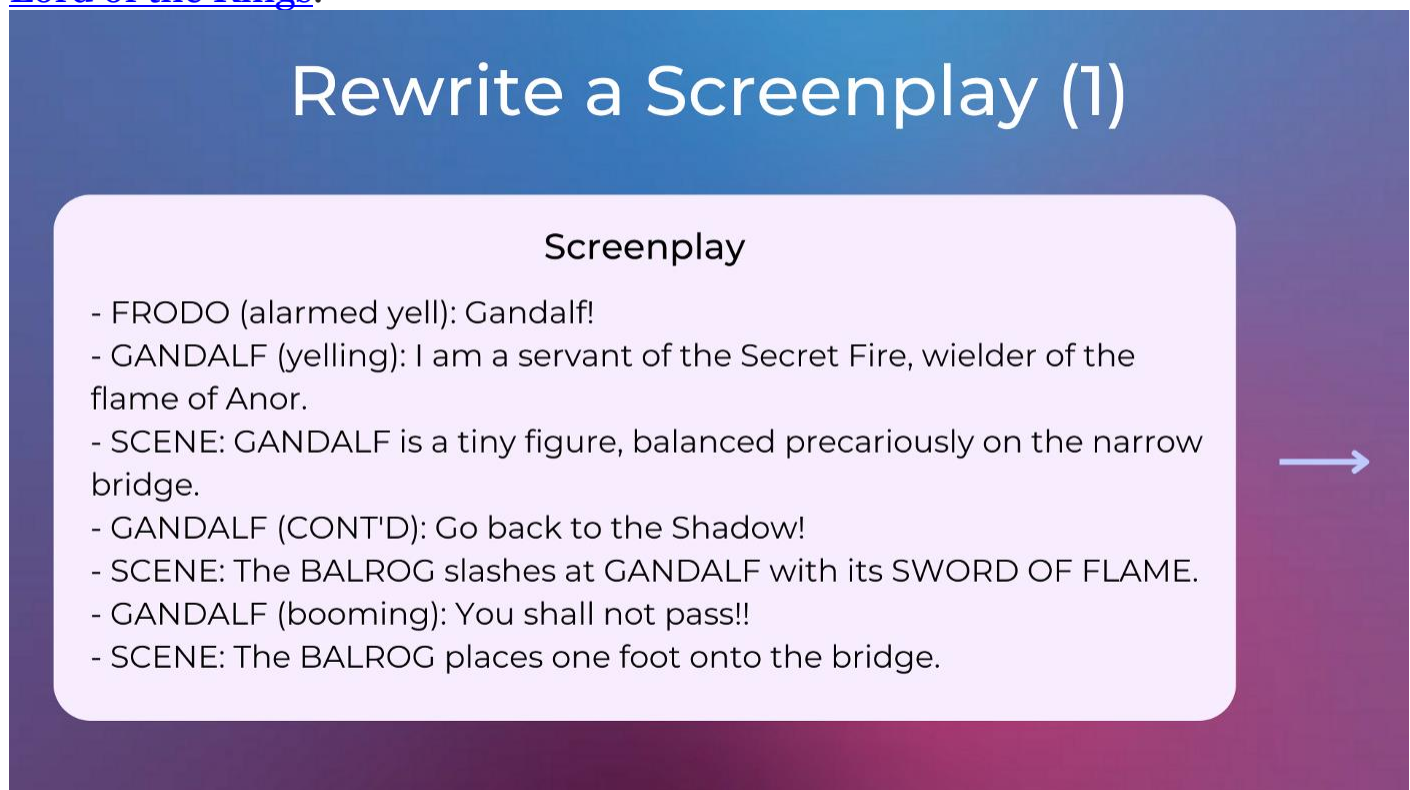
The [Turing Test](#) is a test for determining whether a machine is able to exhibit intelligent behaviour, indistinguishable from what a human would do. It is named after Alan Turing, who proposed it in 1950. The test is simple: a human judge engages in a natural language conversation with two other parties, one of whom is a machine and the other is a human. If the judge cannot tell which is which, then the machine is said to have passed the test.

In an experiment, human were asked to read short articles (about 200 words long) and distinguish whether they were generated by GPT3 or written by a human. As a result, 52% of the times the humans guessed correctly, which is really close to the performance of choosing at random (50%).

*Does this mean that GPT3 passes the Turing Text? No. While GPT3 is very good at generating articles and writing coherent text, it tends to produce inconsistencies in long text. In addition, holding a conversation has numerous additional challenges, as the model must give plausible and coherent responses to a series of messages.*

## Rewriting Screenplays

Similarly, a model of this type is also able to rewrite an existing text following specific instructions. Suppose we want to change the following scene from [The Lord of the Rings](#).



### Rewrite a Screenplay (1)

**Screenplay**

- FRODO (alarmed yell): Gandalf!
- GANDALF (yelling): I am a servant of the Secret Fire, wielder of the flame of Anor.
- SCENE: GANDALF is a tiny figure, balanced precariously on the narrow bridge.
- GANDALF (CONT'D): Go back to the Shadow!
- SCENE: The BALROG slashes at GANDALF with its SWORD OF FLAME.
- GANDALF (booming): You shall not pass!!
- SCENE: The BALROG places one foot onto the bridge.

Image by the author.

We can use as input to the model the screenplay and instructions on what to do with it. For example, let's try to modify this scene to make Gandalf befriend the Balrog (i.e. the monster featured in the famous "You shall not pass" scene), rather than fight him.

# Rewrite a Screenplay (2)

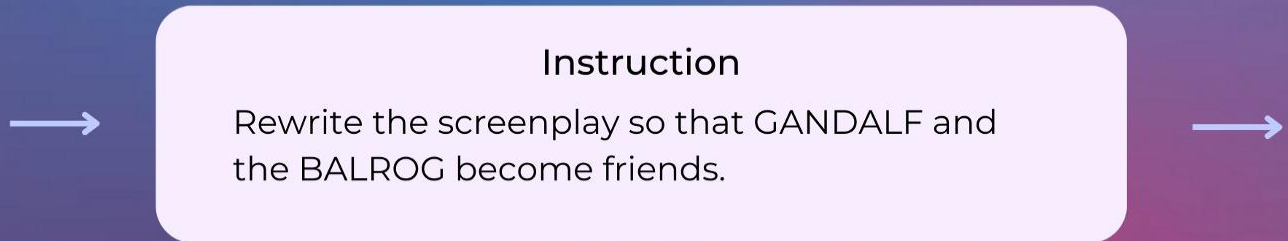


Image by the author.

Here is the revisited screenplay, in which a situation is reached where Gandalf and the Balrog begin to forge a bond. How is it possible? The model simply wrote the continuation of text it deemed most likely, thanks to the knowledge gained from its training data.



# Rewrite a Screenplay (3)

## Screenplay


- 
- FRODO (alarmed yell): Gandalf!
  - GANDALF (yelling): Let's be friends!
  - SCENE: GANDALF is a tiny figure, balanced precariously on the narrow bridge and he holds up a hand to the BALROG.
  - GANDALF (CONT'D): Let's be friends!
  - SCENE: A ringing clash and the BALROG'S SWORD SHATTERS into MOLTEN FRAGMENTS! The BALROG, who has been hit in the face with a piece of molten metal, is crying. GANDALF is rubbing his hand and looks at the BALROG with concern.
  - GANDALF (booming): Are you OK?
  - SCENE: The BALROG places one foot onto the bridge and shakes its head.

Image by the author.

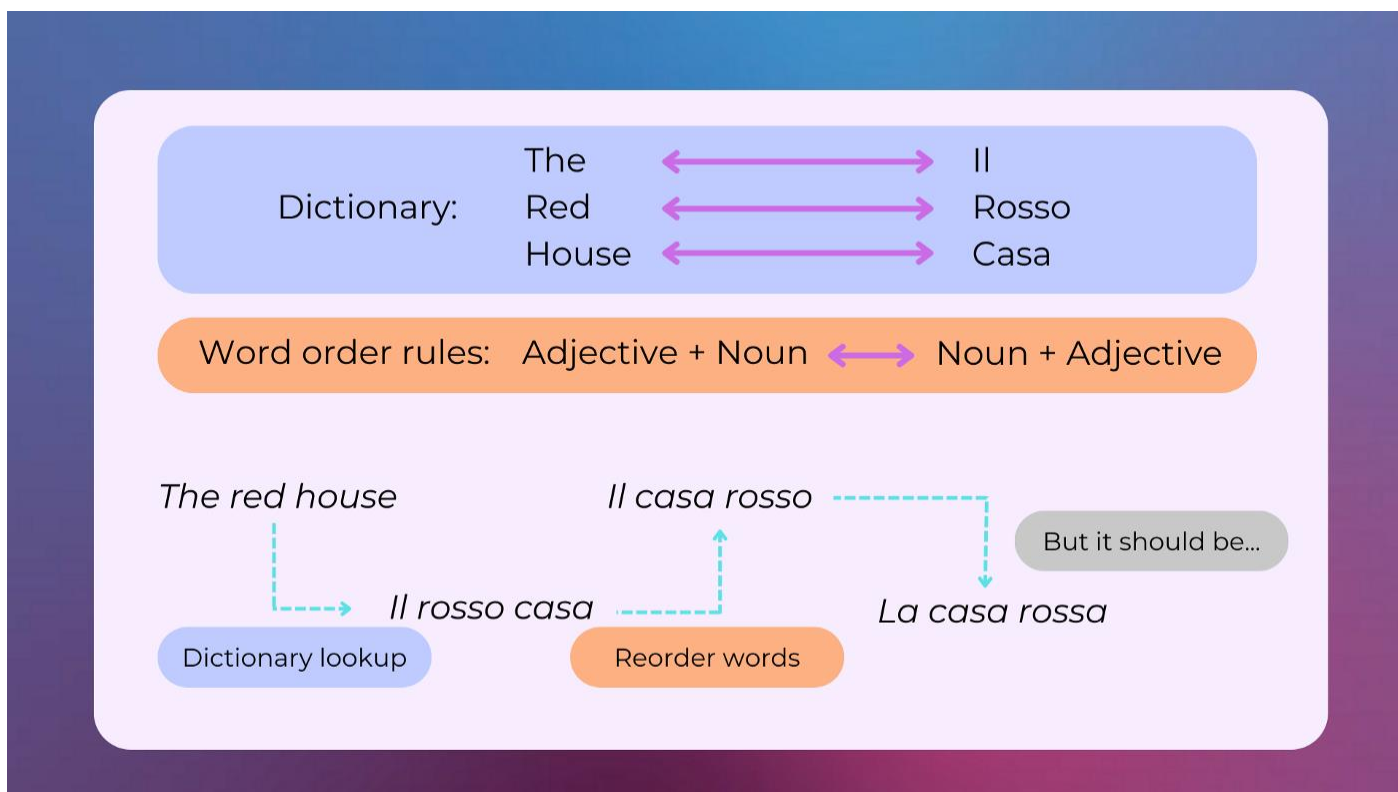
I hope these two examples show the current state of NLP advancement. Obviously, these models have limitations and they are not perfect, not all that glitters is gold. Despite this, results that were unthinkable only 5 years ago can now be achieved, and few people are aware of them.

## A Brief History of NLP

NLP has had a history of ups and downs, influenced by the growth of computational resources and changes in approaches.

### The 1950s, 1960s, and 1970s: Hype and the First AI Winter

The first application that sparked interest in NLP was machine translation. The first machine translation systems were very simple, using dictionary lookup and basic word order rules to produce translations.



How the first machine translation systems worked. Image by the author.

In 1957, the idea of generative grammar was introduced by [Chomsky](#) in the famous book [Syntactic Structures](#), which helped researchers to better understand how machine translation could work.

## ***Generative grammar***

*A [generative grammar](#) is an attempt at creating a set of rules able to correctly predict whether a text is grammatically (with respect to a specific language) correct or not. It's a system of explicit (and usually recursive) rules that can be used to generate all the possible sentences in a language.*

The 1950s saw a lot of excitement around the potential for fully automatic high-quality translation systems. However, this was quickly dampened by the reality that such systems were far from achievable at the time. These systems were not very effective, as they did not take into account the ambiguity of natural language.



In 1966, the [ALPAC](#) (Automatic Language Processing Advisory Committee, a committee established in 1964 by the United States government in order to evaluate the progress in computational linguistics) released a report that recommended research into machine translation be halted, which had a significant impact on research in natural language processing and artificial intelligence more broadly.

### ***The ALPAC report of 1966***

*As a conclusion, the [recommendations](#) of the report were to support machine translation research less on the achievement of a full end-to-end solution and more on being able to aid translators in their work. For example, the report suggested to look for practical methods for evaluating translations, evaluation of available sources of translations, and ways of speeding up the human translation process.*

### ***AI Winters***

*The history of artificial intelligence has experienced several hype cycles, followed by disappointment for not meeting high expectations, followed by research funding cuts, followed by a period of several years of little research (called [AI winters](#)), followed by renewed interest and hype again.*

*The first cycle began with the enthusiasm of the 1950s and ended with the 1966 ALPAC report and the subsequent AI winter until the end of the 1970s.*

In addition to the development of language theories, several prototypes were also created during this period. The most famous prototype is perhaps [ELIZA](#), designed to replicate the conversation between a psychologist and a patient. Here is a sample conversation with ELIZA.

Welcome to

```
EEEEEE LL      IIII  ZZZZZZ  AAAAA
EE      LL      II    ZZ      AA   AA
EEEEEE LL      II    ZZZ      AAAAAA
EE      LL      II    ZZ      AA   AA
EEEEEE LLLLLL  IIII  ZZZZZZ  AA   AA
```

Eliza is a mock Rogerian psychotherapist.  
The original program was described by Joseph Weizenbaum in 1966.  
This implementation by Norbert Landsteiner 2005.

```
ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:
```

Image from <https://en.wikipedia.org/wiki/ELIZA>.

Despite the slowdown in NLP research in the 1970s, there have been developments in new computationally tractable theories of grammar, such as [case grammars](#), [semantic networks](#), and [conceptual dependency theory](#).

## The 1980s and 1990s: Expert Systems and Statistical Models

In the 1980s, symbolic approaches (also called expert systems) were used in NLP, with hard-coded rules and ontologies (i.e. a knowledge base of facts, concepts, and their relationships about a specific domain). Here's an example of a fact stored in an ontology.

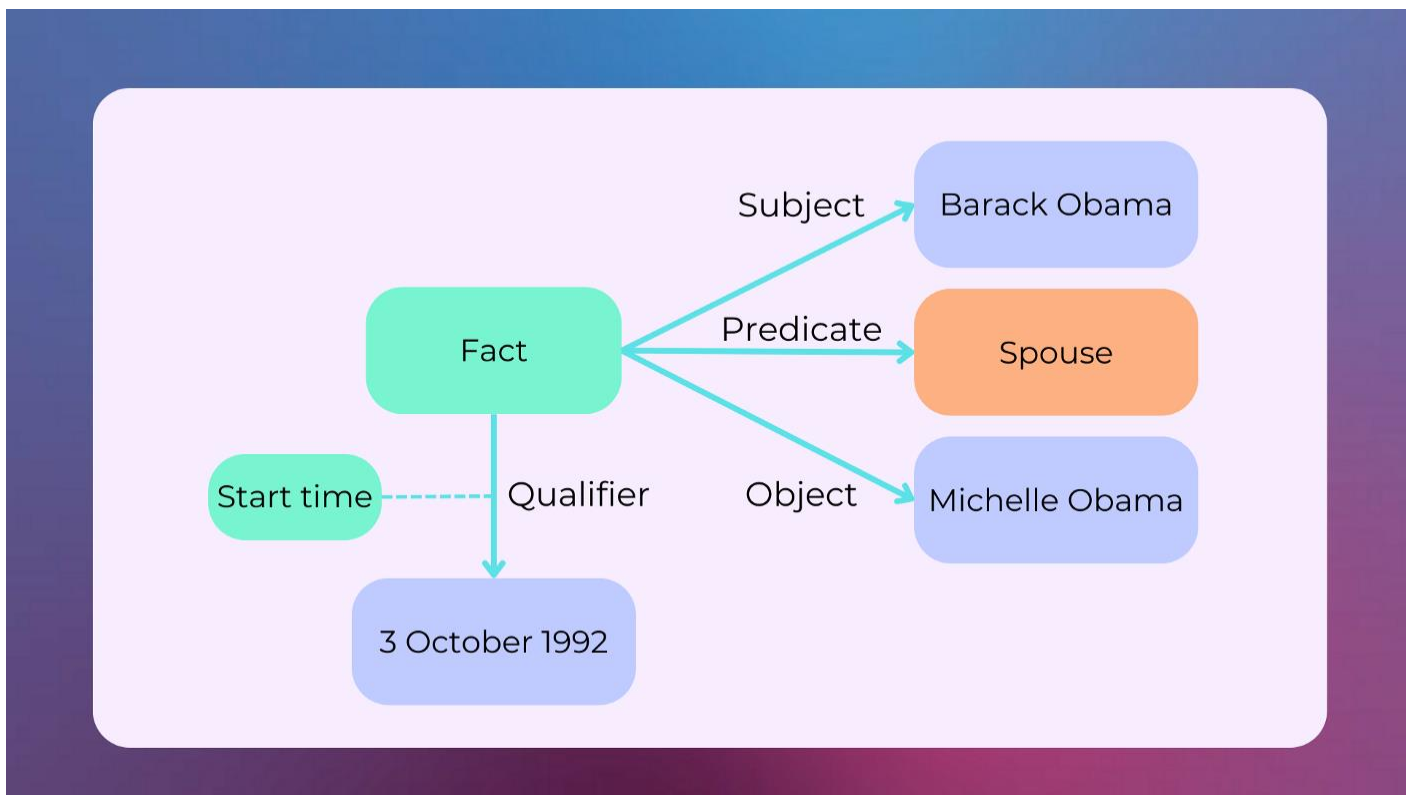


Image by the author.

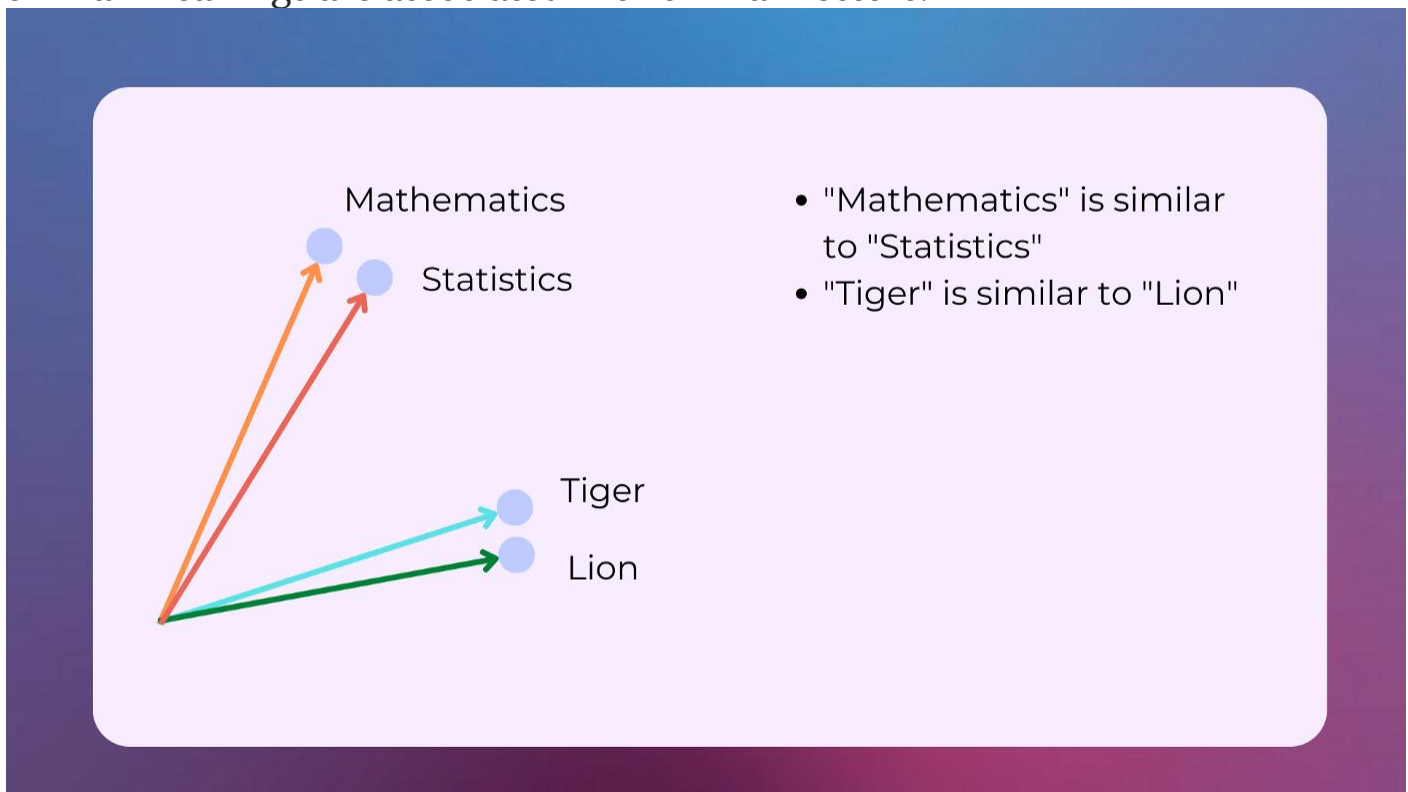
These were programs designed to mimic human experts in specific domains, such as medicine or law. One of the most famous expert systems was [MYCIN](#), which was used to diagnose blood infections.

In the late 1980s and early 1990s, statistical models took over the symbolic approach. Statistical models were able to learn by themselves, through machine learning, the multitude of hard-coded rules of the expert systems. This was made possible above all by the increase in computational resources. The first [recurrent neural networks \(RNNs\)](#) were trained in these years.

## **The 2000s, 2010s, and 2020s: Neural Networks, Embeddings, and Transformers**

In the 2000s, neural networks are used more and more, initially for learning the distribution of language words and predicting the next words in a text given the previous ones (i.e. language modeling). Also in this period, we begin to represent

words with dense vectors of numbers called word embedding, so that words with similar meanings are associated with similar vectors.



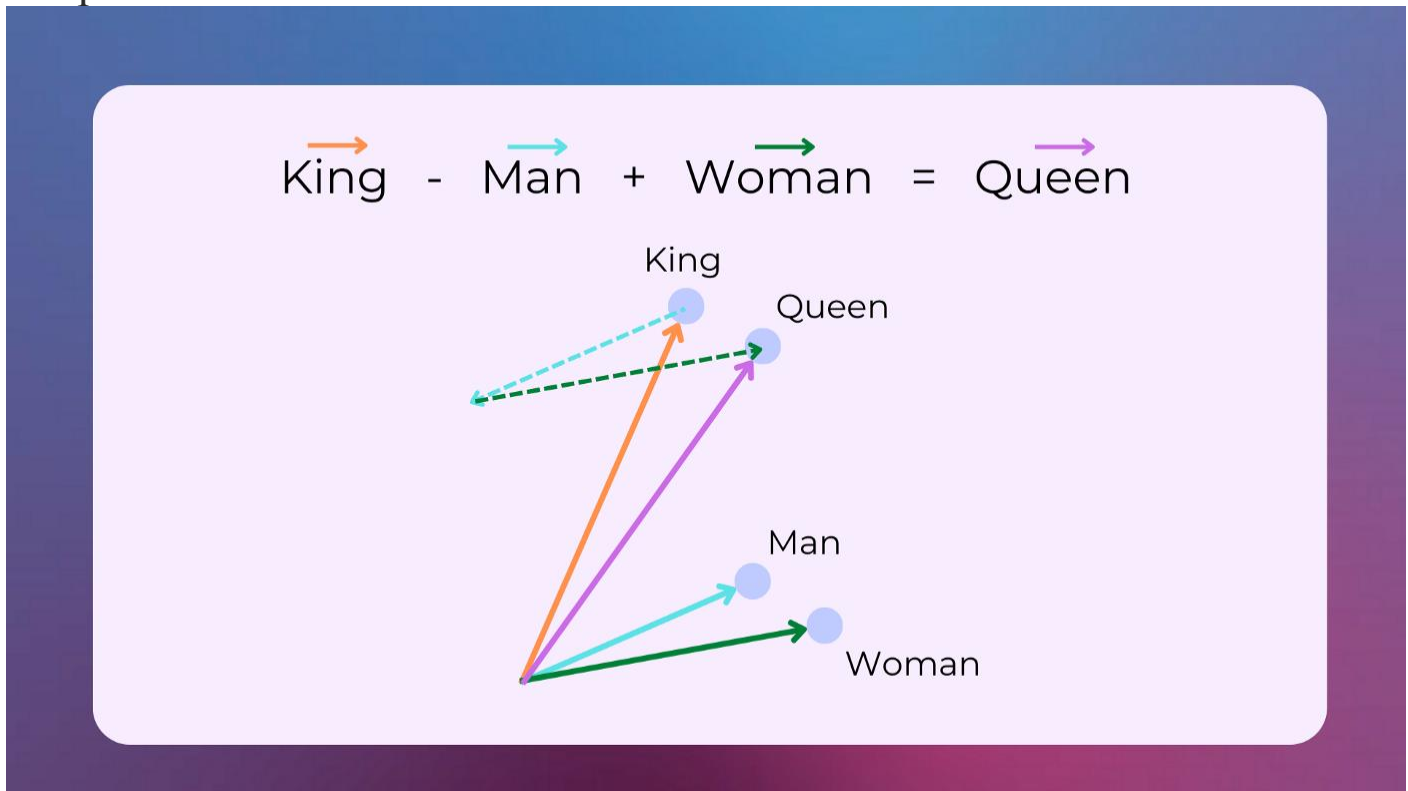
A graphical representation of word embeddings. Image by the author.

However, early algorithms were unable to efficiently learn this representation and so they were trained on a small amount of text, thus producing suboptimal vectors.

The first commercially successful natural language processing system was the [Google Translate service](#), which was launched in 2006. Google Translate used statistical models to automatically translate documents from one language to another.

In 2013 the Word2Vec paper "[Efficient Estimation of Word Representations in Vector Space](#)" was published, i.e. the first algorithm capable of learning word embedding efficiently, thus greatly improving previous word embedding. In the first experiments with these vectors, it was noticed that it is possible to do

mathematical operations with them that represent semantic operations. For example, taking the vector of the word “king”, subtracting the vector of “man” from it and adding the vector of “woman”, we get a vector very close to the vector of “queen”.



A graphical representation of word embedding operations. Image by the author.

What made word embeddings of great importance in NLP until today has been the evidence that using pre-trained embeddings as features for machine learning models improves performance across a wide range of downstream tasks. Since then, a lot of work has been done to obtain word embeddings that better encapsulate the meaning of the texts they represent. The most used neural networks during this period were [LSTM RNNs](#) and [Convolutional Neural Networks \(CNNs\)](#).

Subsequently, in 2014 a general formalization of sequence-to-sequence problems (such as machine translation, which translates a sequence of words into another sequence of words) was proposed with a system made up of two neural networks

called Encoder-Decoder: the first network is called Encoder and encodes the entire input into a vector (commonly called context vector), the second is called Decoder and decodes the output from the vector.

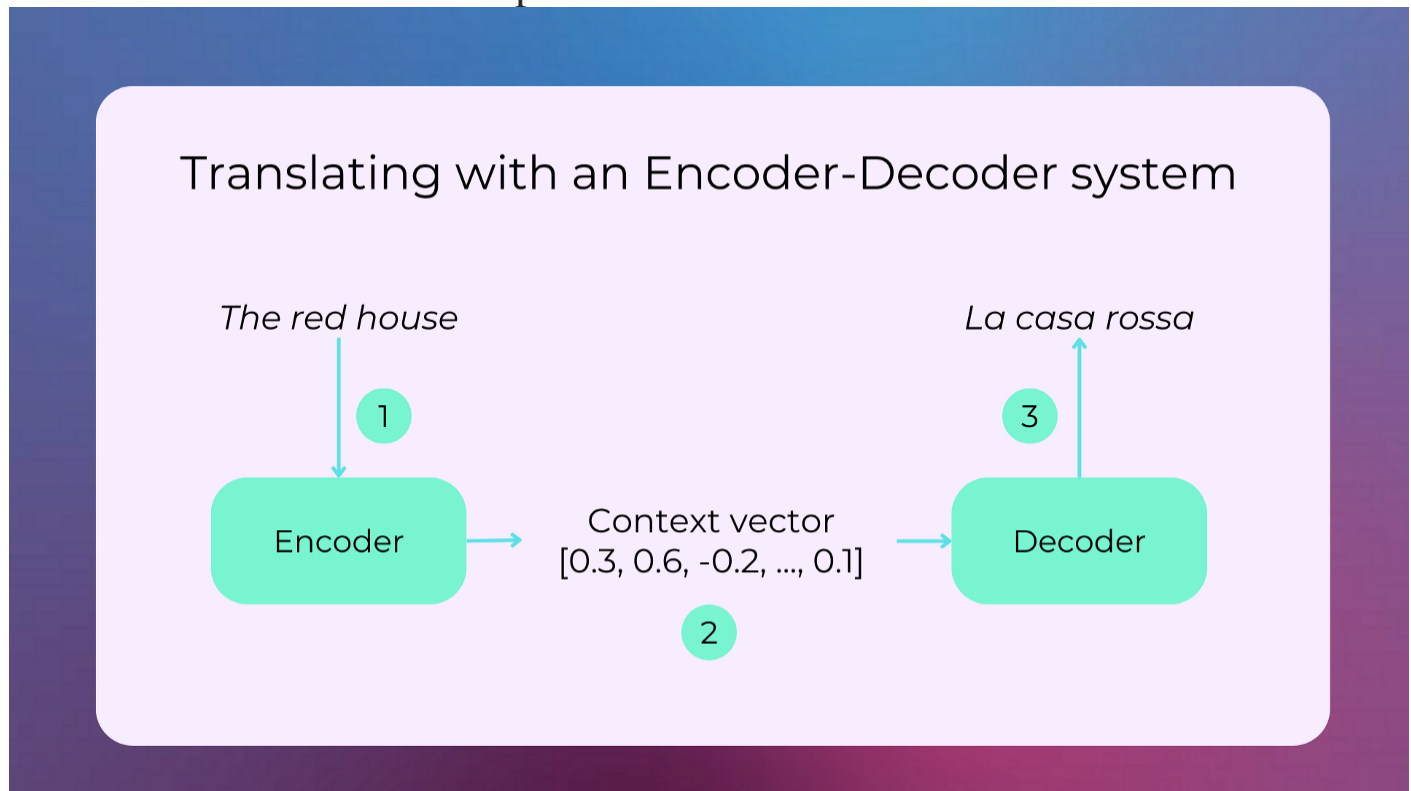


Image by the author.

The encoder-decoder architecture was then very successful with the use of [attention mechanisms](#), which allow a greater passage of information between the two modules. This new system was so successful that, in 2016, [Google Translate replaced its statistical model with a neural sequence-to-sequence model](#). The attention mechanism was then used in 2017 also to create a new type of neural network called [Transformer](#) (with the famous paper [Attention Is All You Need](#)), which is still the most used neural network in NLP today.

In the following years, Transformers were trained in such a way as to be able to produce embeddings from texts that also take into account the context in which the words are, advancing the state of the art in the majority of NLP tasks. These models are essentially language models trained in a self-supervised way on



datasets comprising large quantities of texts from around the Internet. These models are also called pre-trained models because, after being first trained on a large training set, they can then be reused in different tasks after a small adaptation training (called fine-tuning).

A trend today is to train increasingly larger language models on ever larger datasets. The reason for this is that these models are able to solve a wide range of NLP tasks excellently and without further training. The two examples of generating draft articles and rewriting screenplays were produced using one of these large language models.

## **The Recent Boom of NLP**

It's no secret that Natural Language Processing has seen a boom in recent years, especially from Word2Vec onwards. The main reasons for its success are:

- The increasing availability of text data from around the internet, thanks to which it is possible to learn the characteristics of a language starting from billions of sentences.
- The development of powerful computational resources, especially better and better hardware for the type of computations that neural networks need (e.g. GPUs and TPUs).
- The development of frameworks for developing neural networks like TensorFlow and PyTorch.
- The advances in machine learning algorithms, such as representing text with semantic vectors, transformers, and attention mechanisms.