Natural Language Processing Applications in Engineering

Seda YILDIRIM
KTO Karatay University

Introduction

Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence. NLP related to the interactions between computers and human language, how to program computers to process and analyze big amounts of natural language data ("Natural Language Processing", 2022).

NLP brings together theories, methods and technologies developed in many different fields for example artificial intelligence, formal language theory, theoretical linguistics, and computer-aided linguistics. The purpose of NLP is to examine problems in automatically generating and understanding natural languages (Young et al., 2018). NLP helps ensure human-computer interaction by processing human-generated sounds and texts.

NLP are theoretically motivated computational techniques for automated analysis and representation of human language (Cambria et al., 2014). In an environment where people of all ages can access social media, the amount of data generated continues to increase day by day. Naturally generated data by humans is not in a condition to be processed directly. Therefore, many areas need to work together to make sense of data and use it efficiently to provide human-machine communication.

The main objectives of research in the field of NLP are as follows:

- Better understanding of the function and structure of natural languages
- Using natural language as an interface between computers and humans and facilitating communication between computer and human.
- Making language translation with computer

To understand the structure of natural languages, it is necessary to make a detailed analysis of natural language and to extract its mathematics. Natural Language Processing aims to understand or reproduce the canonical structure of natural languages by analyzing them.

History of NLP

The first natural language processing studies started in 1950 with the article Computer Machines and Intelligence published by Alan Turing. In this article, he made a definition that includes automatic interpretation and natural language generation known as the Turing Test ("Natural Language Processing", 2022).

NLP history consists of 3 stages:

- Symbolic NLP (1950s-early 1990s): Rule-based emulation of Natural Language understanding and Generation Labeling operations made according to grammatical

rules. They work according to the grammatical structures of languages. For example, 'x' is an unknown word in English. If there is a sentence in the form of Adverb + x + Noun, the word x is an adjective.

- Statistical NLP (1990s–2010s): Statistics-based methods like bag-of-words and n-grams became popular, also thanks to increased data availability from the Internet.
- Neural NLP (present): Due to the successes achieved in language modeling and parsing studies (Goldberg, 2016, Goodfellow et al., 2016, Jozefowicz et al., 2016, Choe & Charniak, 2016, Vinyals et al., 2014), machine learning techniques such as representational learning and deep neural learning have become widespread in natural language processing since 2010.

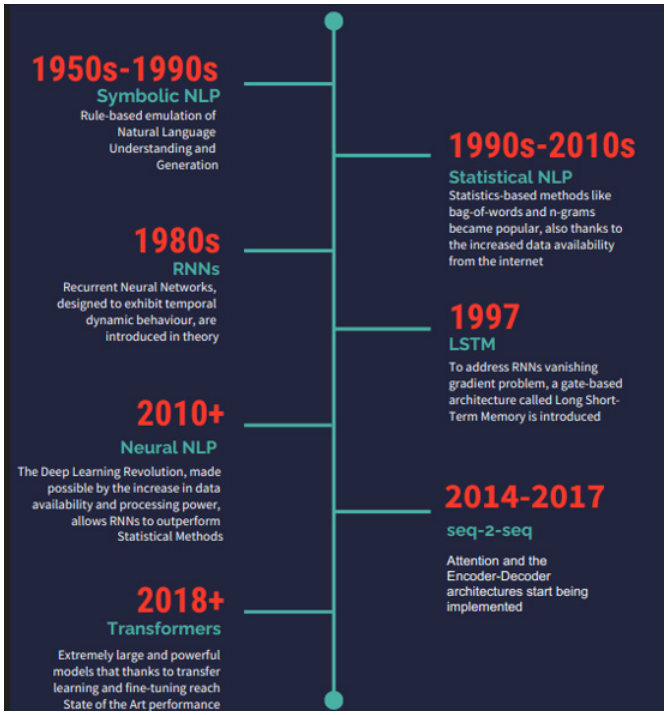

Figure 1. NLP brief history ("Natural Language Processing History", 2022)

Methods

Statistical-Based Methods

The first of the commonly used methods in natural language processing is *statistical-based methods*. Language modeling is the job of finding statistical distributions that capture patterns in natural language. A statistical language model is built on the weights or probabilities of terms. Word representation methods are divided into two as *Frequency Based Representation* and *Prediction Based Representation*. Frequency-based word representations, which are defined as more traditional methods, are focused on the principle of detecting the words in the documents and the frequencies of these words.

Bag of Words Method

The most preferred method developed based on *frequency-based word representation* is the Bag of Words method. According to this method, each sentence in the document is divided into unique words and converted into a unique word-sized matrix. While the columns of the matrix consist of the words in the document (N), the rows consist of the number of the document (D). As a result, the entire corpus is represented as a DxN matrix.

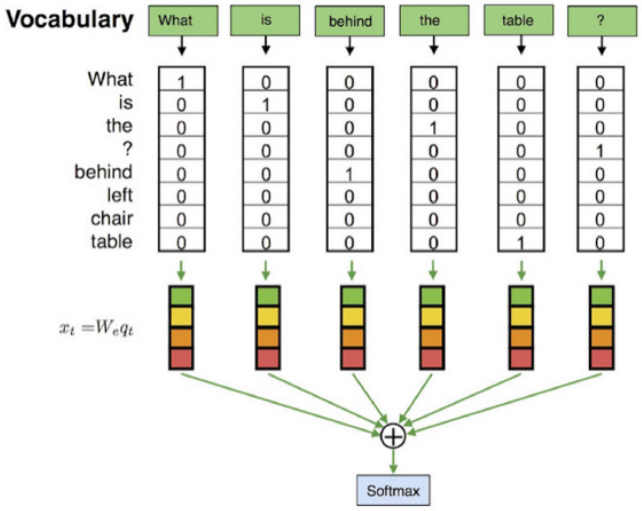

Figure 2. Bag of words example (Malinowski & Fritz, 2016)

Frequency-based methods have two main disadvantages (Karcı & Aydoğan, 2019)

- First, because there are unique words in the rows and/or columns according to the method used, matrices in the size of the word count occur, and since the value of most of them will be 0, a sparse matrix is formed. This causes the matrix to take up space in memory.
- Another important disadvantage is the inability to detect semantic closeness between words. In other words, if a word always comes after a word in a text, the link between it cannot be determined. For example, let's examine the sentence "Ali came to school". When the words "Ali" and "to school" come, the future of the words "came" or "went" should be deduced from the text.

Word2Vec Method

One of the word representation methods, the *prediction-based word representation method*, is the Word2Vec model, which was developed in 2013 and is based on the principle of training words with an artificial neural network. The Word2Vec model consists of two different algorithms named *CBOW (Continuous Bag of Words)* and *Skip-gram.*

The main differences of these two models are based on the methods of taking inputs and outputs. Word2Vec method, takes unique words in a large corpus as input and a matrix

with a specified representation vector size is created. This method scans a sentence in the document each time by scrolling through a structure called a window and produces a vector consisting of many dimensions according to the target word in the window (Karci & Aydogan, 2019).

CBOW Model (Continuous Bag of Words)

In the CBOW model, words that are not in the window size center are taken as input and the words in the center are tried to be estimated as output. This process continues until the end of the sentence. This situation is tried to be shown in Figure 3. The value indicated by $w_{(t}$ is the output value in the center of the sentence and desired to be estimated, while the values indicated by $w_{(t-2)}$…..$w_{(t+2)}$ are off-center output values according to the preferred window size (window size).

Example Sentence: "Artificial intelligence is the new electricity."

The CBOW model, which takes this sentence as input and window size = 1, works as follows. First, the word "artificial" is placed in the center of the window, then the word "artificial" in the center of the window is tried to be predicted with the neural network model by taking the 1 word on the right and the left as separate input (window size=1). Then window 1 is shifted to the right, this time with the word "intelligence" in the center of the window. Table 1 shows the step-by-step operation of the sentence according to the CBOW model. The words written in red represent the output, and the words in blue represent the input (Karcı & Aydoğan, 2019).

Table 1. How the CBOW model works (Karcı & Aydoğan, 2019)

	Sentence (window size=1)	**Input**	**Output**
1	Artificial intelligence is the new electricity.	(intelligence)	(artificial)
2	Artificial intelligence is the new electricity.	(artificial) (new)	(Intelligence) (Intelligence)
3	Artificial intelligence is the new electricity.	(Intelligence) (electricity)	(new) (new)
4	Artificial intelligence is the new electricity.	(new)	(electricity)

Skip- Gram Model

In the Skip-Gram Model, it has a reverse operation of the CBOW model. The word in the center is taken as input and the words that are not in the center are tried to be guessed as output. This process continues until the sentence ends. This situation is shown in Figure 4 (Karci & Aydogan, 2019).

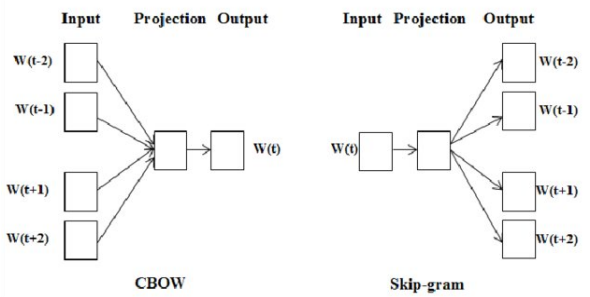

Figure 3. Structure of Skip – Gram and CBOW (Suleiman & Awajan, 2019)

Skip-gram model uses a neural network to create word representations. A common way to represent a word in machine language is to encode a word into an array of characters or a string. However, an array of characters does not carry any meaning. The Skip-gram model will create a vector, an element of vector space to represent each word, called a word vector. A word vector is a position relative to the origin in a graph. Words with similar meaning will have their word vectors closer together whereas the words with different meaning will be far apart. Interestingly, the word vectors encode the semantic relations through linear translations.

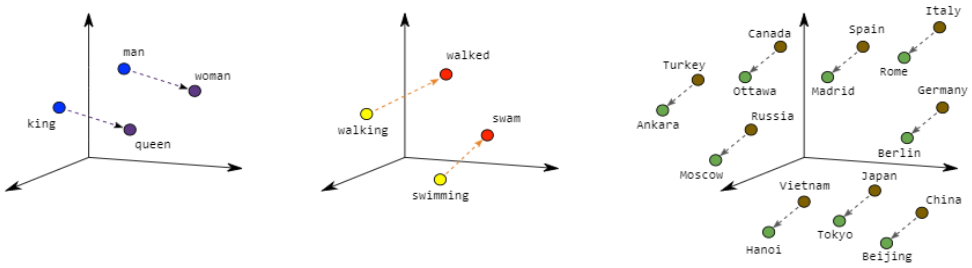

Figure 4. Word Embeddings Representation (Embeddings, 2022)

Artificial intelligence is the new electricity.

We will create (center, context) pairs of words to train our model. The context is the window of words to the left and to the right of a center word. Using a window size of 2,

- Center word, “artificial” will have context words, “intelligence” and “is”.
- Center word, “intelligence” will have context words, “artificial”, “is” and “the”.
- Center word “is” will have context words, “artificial”, “intelligence”, “the” and “new”.
- Center word, “the” will have context words, “intelligence”, “is”, “new” and “electricity”.

Training dataset of (center, context) pairs. The center words will be the input to the neural network whereas context words are the target.

(artificial, intelligence), (artificial, is), (intelligence, artificial), (intelligence, is), (intelligence, the), (is, artificial), (is, intelligence), (is, the), (is, new), etc.

Neural NLP

A major disadvantage of statistical methods is the manual determination of features. Since 2015, statistical methods have been abandoned for natural language processing

and neural networks have been used for machine learning (Socher, 2020) Among the different methods of deep learning, especially *recurrent neural networks* and *convolutional neural networks* give successful results in solving natural language processing problems. For example, the use of convolutional neural networks to solve very different problems of natural language processing has been demonstrated in a study that introduced a unified architecture for multitasking learning (Collobert & Weston, 2008). With these two methods, it has been stated in different studies in the literature that other deep learning methods can be used in various problems of natural language processing and successful systems can be developed based on these methods (Socher et al., 2012).

Deep learning, which is one of the machine learning methods, predicts the results with the given data set and consists of multiple layers. Deep learning, machine learning; machine learning is a sub-branch of artificial intelligence. Deep learning (also deep structured learning, hierarchical learning, or deep machine learning) is a field that includes artificial neural networks and machine learning algorithms with one or more hidden layers. Deep learning can be conducted supervised, semi-supervised or unsupervised. Deep artificial neural networks have also given successful results with the reinforcement learning approach. In order to define a model with classical machine learning techniques, the feature vector must be extracted first. In order to extract the feature vector, the data must be preprocessed. Deep Learning has eliminated the data preprocessing problem that machine learning workers have been dealing with for many years. Because, unlike traditional machine learning and image processing techniques, deep networks perform the learning process on raw data. While processing the raw data, it obtains the necessary information with the representations it has created in different layers.

For deep learning to learn distinctive features by itself, the number of data must be large. The more data there is for the learning process, the more successful the system will be. The data passes through multiple layers, revealing the details on the data.

Three main types of deep web models:

- Convolutional Neural Networks
- Recurrent Neural Networks
- Multilayer Perceptron (Multilayer Perceptron)

Convolutional Neural Networks

The CNN algorithm is very successful in problems such as image classification, object identification and image segmentation. CNNs are enhanced versions of ANNs. CNN is the network that gets deeper as a result of the increase in the number of hidden layers in ANNs. This depth in CNN is achieved using 2D filters. CNNs realize learning in a hierarchical structure (Kucuk & Arici, 2018).

CNN uses convolution and pooling operators. A CNN has three basic types of layers:

- Convolutional layer

- Pooling layer
- Fully connected layer

Multiple convolution + pooling can be done in succession. Then there are several fully connected layers. In multi-label classification problems, there is the SoftMax layer at the end. The fully connected layer receives the three-dimensional input by reducing it to one dimension and obtains a class label. The SoftMax layer calculates the probability tribution of the output classes.

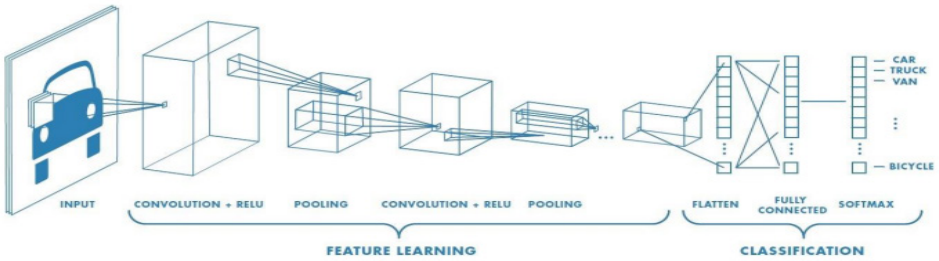

Figure 5. CNN Layers (Akcayol, 2016)

Recurrent Neural Networks

Recurrent neural networks (RNN) are a family of artificial neural networks specialized for processing sequential data. Convolutional neural networks can be scalable on data of large width and height, or they can process data of varying sizes. Similarly, iterative neural networks can scale and process sequential arrays for row-based data larger than non-specialized networks can handle. Recursive networks are based on the idea of sharing parameters between different parts of the model so that the model can be extended and applied to different form instances and generalized (Goodfellow et al., 2018). Although the difference between standard neural networks and iterative neural networks may seem insignificant, the implications of sequential learning of iterative neural networks are far-reaching. A neural network model can only map from input vectors to output vectors, while iterative neural networks can map to any output from the entire history of previous inputs. The equivalent consequence of the universal approximation theory for standard neural networks is that an iterative neural network with enough hidden units can approximate any measurable sequence-to-sequence mapping (Hammer, 2000).

The key here is that recurring connections allow a “memory” of previous inputs to remain in the internal state of the network and thus affect network output. That is, recurrent neural networks have a dynamic nature and backward neurons have backward connections. The output of the network depends on the input values of the network, the previous input values or the parameters such as the previous output values (Geron, 2017).

In summary, there is a loop fed into the RNN. The RNN structure is as follows.

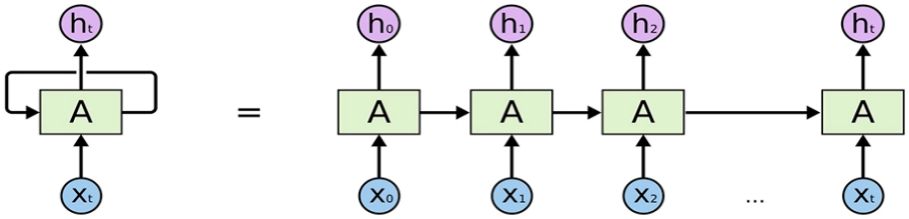

Figure 6. Recurrent Neural Network (An Introduction to Recurrent Neural Networks, 2022)

As a standard, an input comes through the neural network and an output is produced. At the same time, neurons connect with each other over time. In this way, the neural network working now can receive information from the network that worked at the previous time in this way, it can provide a memory flow. For example; The network takes a word and produces any output. When the same network moves to the next word and takes that word as input, the neural network that worked at the previous time also returns a vector, so the neural network can also access information about the previous word. In RNN, the neural network has a memory and information flows over time. Since the network can receive information from networks that have worked in the previous steps, it can interpret the sequential data in a healthier way.

Different RNN Architectures

One to one: It is the standard neural network architecture. It takes an input of a certain size and returns an output of a certain size. For example; image classification

One to many: It is the classical RNN architecture. For example; In automatic image captioning, an image is given as input, and words describing what is happening in this image are taken as output. It could also be the other way around.

Many to one: Give more than one input and get a single output. For example; In Sentiment Analysis, a text is given to the neural network and it is determined whether this text is positive or negative.

Many to many: Different numbers of inputs are given, and different numbers of outputs are received. For example; machine translation and Chatbots

An output is produced for each processed input that differs from the other. In the former, the input is processed first and then the output is produced. For example; The input can be a video and it is desirable to make a video in which every frame of the video is tagged.

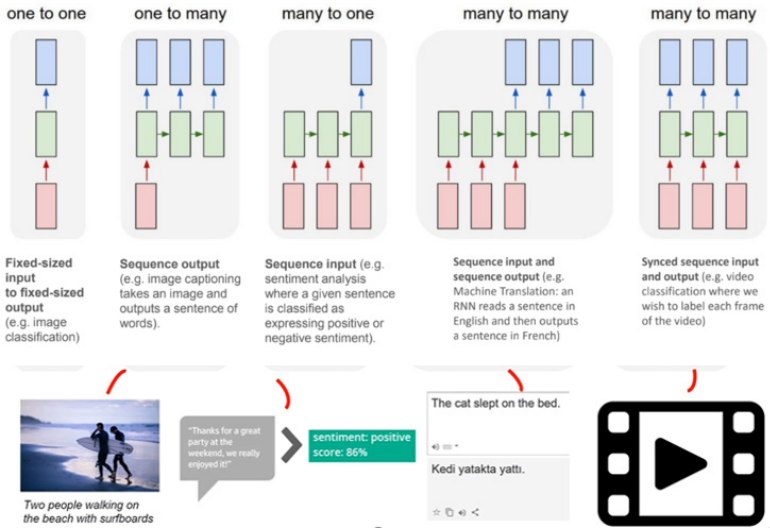

Figure 7. Different RNN architectures (Seq2Seq models,2019)

NLP Studies

The studies in the field of Natural Language Processing and the Deep Learning methods used in these studies are as follows.

Table 2. Natural Language Processing Problems and Deep Learning Methods Used (Socher et al., 2012)

Natural Language Processing Problem	Deep learning methods used
Text Classification	Convolutional Neural Networks Recurrent Convolutional Neural Networks Long Short-Term Memory
Text Parsing	Convolutional Neural Networks
Sentiment Analysis	Deep Auto Encoders Convolutional Neural Networks
Information Extraction	Deep Neural Networks
Asset Name Recognition	Convolutional Neural Networks
Temporal Relationship Inference	Convolutional Neural Networks
Event Inference	Convolutional Neural Networks
Vocabulary Labeling	Deep Neural Networks Long Short-Term Memory
Text Sorting	Convolutional Neural Networks
Automatic Transliteration	Deep Belief Networks
Automatic Question Answering	Convolutional Neural Networks Long Short-Term Memory

(*) LSTM is an artificial iterative neural network (RNN) architecture used in the field of deep learning. Unlike standard feed-forward neural networks, LSTM has feedback connections.

Text Classification: One of the long-studied and important application areas of natural language processing is text classification. Deep learning methods are also used for text

classification and there are various current studies on this subject. In a sample study, very successful results were obtained with a simple single-level convolutional neural network approach for the sentence classification problem (such as positive/negative customer reviews) on different datasets (Kim, 2014). In another case study, an iterative convolutional neural network infrastructure is proposed for text classification. In the proposed approach; Iterative neural network is used for contextual information and convolutional neural network is used for text representation (Lai et al., 2015)

Text Parsing: In text parsing, it is aimed to reveal the grammatical structure of a given text. In a study on this subject, a fast method based on convolutional neural networks has been proposed for text parsing. In this method, convolutional neural networks are combined with label extraction structured in graphs, and thus the graph converter network method, which was previously available in the literature, was used (Collobert & Weston, 2008).

Sentiment Analysis: Sentiment analysis is defined as the automatic determination of emotion, opinion and subjectivity in each text (Chen et al., 2015). Commonly, using the text given in sentiment analysis studies, it is aimed to classify this text as positive, negative or neutral. Today, the widespread use of the Internet to enter all areas of life provides people with virtual environments such as social media, forums, blogs, e-commerce sites, where they can share their ideas, feelings and opinions. In these virtual environments, there is a huge amount of data in which people's opinions are expressed. These data attract the attention of many people and researchers, especially the manufacturers and sellers of these products and services. Idea mining and sentiment analysis aims to reveal the feelings, ideas and thoughts hidden in the texts in which people express their views on topics such as products, services, organizations, events, political thoughts in virtual environments.

Information Extraction: Information extraction from texts in natural language; Many important pieces of information such as entities, events, date and time expressions, concepts are automatically extracted. *Entity name recognition* is a sub-problem of the information extraction field of natural language processing and is a long-studied research topic. Entity name recognition systems, in general, automatically extract and classify the names of entities such as people, places and institutions in each text. *Inference of temporal relations* is also a sub-topic of information extraction and thus an important research area of natural language processing. Temporal relations include the following and similar expressions: "before, after, just before, immediately after, during (during)". Automatic extraction of events mentioned in natural language generated texts is also a sub-field of information extraction and thus a problem of natural language processing (Kucuk & Arici, 2018). In a recent study in this field, an approach using convolutional neural networks is introduced for the extraction of sentence-level features for automatic event extraction. In this study, a different word representation approach was adopted for

the used word-level attributes. This study, which uses deep learning methods for event extraction, has been shown to give successful results in event extraction experiments.

Natural Language Processing and Infrastructure

NLP studies first go through a preprocessing that extracts features from audio or text to understand the text.

Preprocessing steps in natural language processing can be examined under 4 main headings: (Delibaş, 2008)

- Phonology: It examines the sounds of letters and how they are used in language. All languages have an alphabet, and each letter sounds different from the others. The aim of phonology is to translate spoken language into written language. Sounds are tried to be made into words.
- Morphology: At this stage, the words are handled individually and the structure of the word is examined in accordance with the rules of the language. At the end of this study, every part of each word is analyzed. Suffixes, roots, rules about them and classification of these structures are handled within the scope of morpheme. Morphology is to define the types of elements that make up the form in the language and to classify the formal elements called grammatical rules. The word structure of Turkish is realized by adding derivation and inflection suffixes to roots as suffixes.
- Syntax: Examines how words must be arranged to form sentences. At this stage, the words whose analysis has been completed are combined to form sentences and texts, which are larger elements of the language.
- Semantic: Understanding the sentence structure and taking action as a result happens at this stage. The basic function is to examine the meanings that the sequence of words in the language give to the sentences and to give meaning in this way.

Basic Elements of Natural Language Processing

Since the purpose of natural language processing is to communicate with the computer in natural language, the computer needs to learn natural language rules. For this, the computer needs a general dictionary and various algorithms to use this dictionary. In addition to the general information about the language, the computer also needs a field or task-specific knowledge base that it needs and must be perceived independently of the general structure of the language. There are generally five basic elements in a natural language processing system. These are *parser, dictionary (lexicon), understander, knowledge base* and *generator*. (Delibas, 2008)

Parser: The parser syntactically analyzes the given sentence and builds the parser tree. One of the most widely recognized approaches in parsing is phrase-structure grammars.

This approach is based on Chomsky's generative transformational grammar theory. It aims to group sentences by dividing them into groups. According to this approach, the basic and constitutive unit of the language is the sentence. The sentence consists of two basic structures, the noun phrase and the verb phrase. These clusters are also divided into smaller clusters within themselves. After the parsing process, the words whose tasks are determined are subjected to semantic analysis and an output sentence is formed according to the input sentence.

Dictionary (Corpora): It is a structure that contains all the words required to be recognized by the program. The parser works by doing syntactic analysis with the dictionary. The dictionary contains the root and meanings of each word that are required to be recognized by the natural language processing system. It is a work that gives the vocabulary of a language or languages with the way they are spoken and written and shows the words and meanings they form with other elements, and their different uses, based on the root of the word.

Understander: Tries to determine what the sentence means with the knowledge base.

Knowledge base: Conceptually, it consists of two sub-components: general knowledge base and task dependent knowledge base. The main task of the understander is to find the equivalent of the generated parser tree in the knowledge base. The interpreter prepares the appropriate answer for the entered sentence.

Generator: It is the most basic system used in the field of natural language processing and is the display of certain stored patterns for certain words and sentences to the user (Delibaş, 2008).

Natural Language Processing Libraries

Natural Language Processing is a field that produces solutions to many different problems in many different fields. That's why there are different libraries for working in the field of NLP. Python language is commonly used for natural language processing operations. Many reasons such as simple syntax, transparent semantics, open source code, community support are enough to choose Python language. The reason why it is preferred in the field of Natural Language Processing is that it has useful tools for operations such as machine learning and deep learning (Ozen, 2021). These libraries are; *Language Toolkit (NLTK), GENSIM, TORCHTEXT*, *TEXTBLOP*, *CORENLP, ZEMBEREK, ITU Turkish Natural Language Processing Software Chain, SentiTurkNet*

ZEMBEREK: Zemberek is an open source Turkish Natural language processing library and OpenOffice is a LibreOffice extension. Developed entirely in Java, the library has functions such as spell checking, suggestion for incorrect words, hyphenation, and faulty coding cleaning.

ITU Turkish Natural Language Processing Software Chain: It is a work that is not open source for all-natural language processing but supports natural language processing

projects with API support. This study is a platform consisting of tools such as Turkish character converter (deasciifier), separation into sentence elements, spell checker, entity name recognition. Since this platform has both a web interface and an application interface, researchers at different levels can benefit from this platform. There is a preview where we can apply natural language processing methods (Eryigit, 2014).

SentiTurkNet: It is an open source Turkish natural language processing library developed especially for sentiment analysis method. In this library, which was developed by Dehkharghani et al., 2015, a dictionary indicating the emotion poles of Turkish words was created. The classification of the text in this dictionary is made according to six emotions called anger, hate, fear, happiness, sadness and surprise.

Machine learning libraries are also used in natural language processing, which is a part of the world of artificial intelligence and data science. For example, while *Scikit Learn* is frequently used in natural language processing projects, deep learning libraries such as *Tensorflow*, *Keras*, and *Pytorch* are among the libraries used to take natural language processing to the next level with the development of methods and possibilities. It is also used to visualize output with visualization libraries such as *Matplotlib*, *Seaborn* and *Bokeh*. The following table lists commonly used NLP libraries for natural language processing problems.

Table 3. Natural Language Processing Method and Libraries (Yılmaz & Yumuşak, 2021)

Natural Language Processing Methods	Libraries
Named Entity Recognition	NLTK, Spacy, AllenNLP, Stanford-NLP, Zemberek
Text Classification	Zemberek, Flair, PyTorch
Language Identification	TextBlob, Zemberek
Vectorization	Gensim, Scikit-Learn
Separating sentence into elements, finding the root of words, finding the head word	Zemberek, Turkish NLP, Turkish Stemmer, Turkish Lemmatizer, Zemberek Parser
Morphology	Zemberek, ITU Turkish NLP Pipeline
Normalization	Zemberek, Fast.ai, Pyspellchecker
Sentiment Analysis	SentiTurkNet, TextBlob

Natural Language Processing Steps

Segmentation of Sentence (Tokenization): It is the process of breaking down sentences in the text into meaningful small unit particles. As a result of the decomposition of the sentence, too many unit particle tokens are formed. Removal of ineffective words (stop word), removal of misspelled words, stemming and lemmatization can be performed as methods to reduce these particles. Tokens are meaningful small units, symbols, words, phrases can be given as examples of tokens.

Word Tokenizer: Splits the sentence into words and extracts the punctuation marks, also separates the possessive "apostrophe s" in English together.

Sentence Tokenizer: Splits the paragraph into sentences.

Treebank Word Tokenizer: Separates words in sentences according to spaces and punctuation marks.

Word Punct Tokenizer: Extracts punctuation marks from the sentence.

Extraction of ineffective words (stop word): It is one of the methods used to preprocess texts. Stop words are generally unnecessary words. Words that are used to improve the flow of sentences but do not make sense when analyzing data are called stop words. They are words like "a, an, the, and, I, me, myself" in English.

Words like these mean nothing. They are important words for understanding sentences, but they mean nothing to the machine when analyzing data. If the maximum number of words in a text is calculated, the words in the first 10 are made up of stop words. Therefore, these words in the text unnecessarily can be removed or a threshold value can be determined so that it does not affect the text to be learned too much.

Finding the root of words: There are 2 methods to find the roots of words: *Stemming* and *Lemmatization*. Both are methods that try to find the root of the words by discarding the suffixes. The method of finding the root of words (stemming) is simpler than finding the root word (lemmatization), and a strong grammatical knowledge is required for algorithms to find the root word (Yilmaz & Yumusak, 2021). The words highlight, highway and high have the same roots and are high.

Table 4. Stemming and Lemmatization difference

Words	Stem roots	Lemma roots
Drive	Drive	drive
Driving	Drive	driving
Driver	Driver	driver
Drives	Drive	drive
Drove	Drove	drove
Cats	Cat	cat
Children	Children	child

Part of Speech Tagging:It is the tagging of words in texts with their elements as syntax. It is the process of putting the class as a label, whichever class belongs to the word, such as noun, verb, adjective, conjunction.

1) Normalization: Text normalization is the name given to the methods used to reduce clutter in the analyzed text. Case adjustments, unnecessary spacing or character removal, number/text corrections, abbreviation corrections, etc. includes regulations. Statistical methods and the use of some distance measuring methods are common in this field. The most well-known method is

the Levenshtein distance method (Yılmaz & Yumusak, 2021).

2) Morphology: It is the process of classifying sentence components that are parsed as tokens according to grammar.
3) Named Entity Recognition: It is the process of defining predefined categories such as person, place, organization, institution through text documents.
4) Vectorization: The first step for the computer to understand the language is to understand the words. For words to be understood, they must be symbolized in a way that the computer can understand, that is, they must be represented numerically. When words are represented numerically, mathematical operations can be performed on words. Vectors are used to represent words numerically. The conversion of texts into numerical expressions is called word embedding.

Methods that allow words, sentences, or documents to be expressed as vectors are called vectorization methods. Since machine learning algorithms or mathematical models work with numbers, textual expressions need such a transformation.

Developed methods for determining grammatical features in written texts are basically grouped into 2 groups:

- Rule-based methods
- Methods based on statistics / probability

These methods are described in the Methods section.

5) Text Classification: Natural language processing applications that predict which subject/class the content of a text belongs to are called text classification applications. There are also various applications that help us decide who is the author of a text or whether an email is spam based on its content.

Sentiment Analysis

Sentiment analysis is defined as the automatic determination of emotion, opinion, and subjectivity in each text (Chen et al., 2015). Commonly, using the text given in sentiment analysis studies, it is aimed to classify this text as positive, negative, or neutral. Idea mining and sentiment analysis aims to reveal the feelings, ideas and thoughts hidden in the texts in which people express their views on topics such as products, services, organizations, events, political thoughts in virtual environments.

Since the early 2000s, sentiment analysis has become a very active field of study as a sub-branch of natural language processing. Sentiment analysis studies are carried out in the fields of data mining, web mining and text mining as well as natural language processing (Liu, 2012). The term sentiment analysis was first used by Tetsuya and Jeonghee in 2003, and the term opinion mining was first used by Kushal et al., in 2003. Although these statements emerged in 2003, studies on this subject started in previous years. The studies of Vasileios and Janyce in 2000, Tong et al., in 2001, Turney in 2002, Pang et al., in 2002 can be cited as examples of the first studies in this area.

Sentiment analysis and opinion mining are the studies that reveal the feelings, ideas and

thoughts hidden in the texts in which people express their views on different topics such as products, services, organizations, events. In sentiment analysis studies, things such as products, services, events, people, about which opinions are expressed, are called assets. Expression of emotion may be about the entity itself, or it may be about an aspect or feature of the entity (Ozyurt & Akcayol, 2018).

(v, ö, d, n) → v: existence, ö: property, d: emotion, n: subject

The main tasks in sentiment analysis are to extract some or all of these four components of emotion expression from texts according to the scope and level of the study.

Existence: The task of extracting entities from the text is not performed in all sentiment analysis studies. For example, in product reviews or product comments on e-commerce sites, an opinion is expressed about a certain entity, so there is no need for entity extraction from the text. On the other hand, in a column about politics, the author can express his opinion about different political parties and politicians. Asset extraction is important in the idea of mining work to be done in such texts. Entity extraction is a rule-based entity extraction task. There are studies in Turkish in this field by Dalkilic et al. in 2010, Seker and Eryigit in 2012, Kucuk and Yazici in 2009.

Property: Whether feature extraction is performed depends on the level of sentiment analysis.

Emotion: This is also called emotional polarity. Emotion polarity is positive or negative. Although some studies have also been classified as neutral, it is not very common. In addition to the positive/negative binary classification, emotion rating studies are also carried out to determine the positivity/negative level of the emotion. Removing emotion polarity is the main task in sentiment analysis and opinion mining.

Subject: The subject can be a person as well as a legal person. In forums, blogs and e-commerce sites, the subject is the user who writes the message, except in very exceptional cases. In texts such as newspaper news in which the ideas and thoughts of people such as administrators and politicians are conveyed, the subject can be third parties instead of the author himself (Ozyurt & Akcayol, 2018).

References

Akcayol, M. Ali, (2016), *Deep Learning* [PowerPoint slides]. SlideShare. https://w3.gazi.edu.tr/~akcayol/files/DL_L6_CNNs.pdf

An Introduction to Recurrent Neural Networks, (2019), https://datascience.eu/machine learning/an-introduction-to-recurrent-neural-networks/.

Cambria, E., White, B., 2014, Jumping NLP Curves: A Review of Natural Language Processing Research, IEEE Computational Intelligence Magazine, 9(2), 48-57.

Chen, Y., Xu, L., Lıu, K., Zeng, D., Zhao, J., 2015, Event extraction via dynamic multi-poolin convolutional neural networks, Annual Meeting of the Association for Computational

Chen, Y., Xu, L., Lıu, K., Zeng, D., Zhao, J., 2015, Event extraction via dynamic multi-pooling convolutional neural networks, *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processin*g, 167-176.

Choe, D. K., Charniak, E., (2016), Parsing as Language Modeling. classification, AAAI Conference on Artificial Intelligence, 2267-2273.

Collobert, R., Weston, J., 2008, A unified architecture for natural language processing: Deep neural networks with multitask learning, *International Conference on Machine Learning (ICML),* 160-167.

Dalkilic, F.E., Gelisli, S., Diri B., 2010, Named Entity Recognition from Turkish texts, IEEE 18.Signal Processing and Communication Applications Congress, Diyarbakır, Türkiye, ss. 918-920.

Dehkharghani, Yucel, R.S., Yanikoglu, B., Oflazer, K., 2015, SentiTurkNet: a Turkish polarity lexicon for sentiment analysis, Language Resources and Evaluation. 50. 10.1007/s10579-015-9307-6.

Delibas, A., 2008, Turkish Spell Check With Natural Language Processing, (Master Thesis, Istanbul Technical University, Institute of Natural and Applied Sciences), İstanbul.

Embeddings: Translating to a Lower-Dimensional Space, (2022, August 12), https://developers.google.com/machine-learning/crash-course/embeddings/translating-to-a-lower-dimensional-space

Eryigit, G., 2014, ITU Turkish NLP web service, *In Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 1-4).

Geron, A., 2017, Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly Media.

Goldberg, Y., (2016), A Primer on Neural Network Models for Natural Language Processing, Journal of Artificial Intelligence Research 57: 345–420.

Goodfellow, I. Bengio, Y., Courville, A., 2018, Derin Öğrenme, Buzdağı Yayınevi.

Goodfellow, I., Bengio, Y., Courville, Aaro (2016), Deep Learning MIT Press.

Hammer, B., 2000, On the approximation capability of recurrent neural networks, Neurocomputing, 31 (1-4), 107-123, (2000).

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., Yongh, W., (2016), Exploring the Limits of Language Modeling.

Karci, A., Aydogan, M., (2019), Investigation of Word Similarities with Word Embedding Methods, Cukurova University Journal of the Faculty of Engineering and Architecture, 34(2),. 181-195.

Kim, Y., 2014, Convolutional neural networks for sentence classification.

Kucuk, D., Arici, N., 2018, A Literature Study On Deep Learning Applications In Natural Language Processing, International Journal of Management Information Systems and Computer Science, 2(2):76-86.

Kucuk, D., Yazici, A., 2009, Named Entity Recognition Experiments on Turkish Texts, In Proceedings of FQAS-2009, 8th International Conference on Flexible Query Answering Systems, *Roskilde*, Danimarka, ss. 524-535.

Kushal, D., Steve, L., Pennock, D.M., 2003, Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews, In Proceedings of WWW'03, 12th International Conference on World Wide Web, Budapest Congress Centre, Macaristan, ss. 519-528, 2003.

Lai, S., Xu, L., Liu, K., Zhao, J., 2015, Recurrent convolutional neural networks for text Linguistics and International Joint Conference on Natural Language Processing, 167-176.

Liu, B., 2012, Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers.

Malinowski, M., Fritz, M., (2016), Tutorial on Answering Questions about Images with Deep Learning.

Natural Language Processing History, (2022, August 12), https://towardsai.net/p/nlp/conditional-story-generation%E2%80%8A-%E2%80%8Apart1

Natural Language Processing, (2022, August 12), https://en.wikipedia.org/wiki/Natural_language_processing

Ozen, M.S., (2021), Python Doğal Dil İşleme Kütüphaneleri, https://www.datasciencearth.com/dogal-dil-isleme-1-4-python-dogal-dil-isleme kutuphaneleri/

Ozyurt, B, Akcayol, M.A., 2018, A Survey On Sentiment Analysis And Opinion Mining, Methods And Approaches, Selcuk Univ. J. Eng. Sci. Tech., v.6, n.4, pp. 668-693, 2018.

Pang, B., Lee, L., Vaithyanathan, S., 2002, Sentiment Classification Using Machine Learning Techniques, In Proceedings of EMNLP-2002, Conference on Empirical Methods in Natural Language Processing, University of Pennsylvania, PA, ABD, ss. 79-86.

Seker, G.A., Eryigit, G., 2012, Initial Explorations on using CRFs for Turkish Named Entity Recognition, In Proceedings of COLING 2012, *24th International Conference on Computational Linguistics*, IIT, Bombay, Hindistan, 2459–2474.

Seq2Seq models and the Attention mechanism, (2019), https://mett29.github.io/posts/2019/12/seq2seq_and_attention/.

Socher, R., (2020), Deep Learning For NLP-ACL 2012 Tutorial.

Socher, R., Bengio, Y., Manning, C. D., 2012, Deep learning for NLP (without magic), Tutorial Abstracts of ACL, 5.

Süleiman, D., Awajan, A., 2019, Comparative Study of Word Embeddings Models and Their Usage in Arabic Language Applications.

Tetsuya, N., Jeonghee, Y., 2003, Sentiment Analysis: Capturing Favorability Using Natural Language Processing, In Proceedings of KCAP-03, *2nd International Conference on KnowledgeCapture*, Sanibel Island, FL, ABD, ss. 70-77.

Tong, R.M., 2001, An Operational System for Detecting and Tracking Opinions in On-Line Discussion, In Proceedings of SIGIR 2001 Workshop on Operational Text Classification, New Orleans, Louisiana, ABD.

Turney P.D., 2002, Thumbs up or Thumbs down: Semantic Orientation Applied to Unsupervised Classification of Reviews, In Proceedings of ACL'02, *40th Annual Meeting of the Association for Computational Linguistics*, Pennsylvania, ABD, ss. 417-424.

Vasileios, H., Janyce, M.W., 2000, Effects of Adjective Orientation and Gradability on Sentence Subjectivity, In Proceedings of COLING-2000, *18th Interntional Conference on Computational Linguistics,* Saarbrücken, Almanya, ss. 299-305.

Vinyals, O., Kaiser, L., Koo, T., Petrov, S., Sutskever, I., Hinton, G., (2014). Grammar as a Foreign Language, NIPS, http://arxiv.org/abs/1412.7449

Yilmaz, H., Yumusak, S., 2021, Open Source Natural Language Processing Libraries, Istanbul Sabahattin Zaim University Journal of the Institute of Science and Technology,3 (1): 81-85.

Young, T., Hazarika, D., Poria, S., Cambria, E., (2018), Recent Trends in Deep Learning Based Natural Language Processing. IEEE Computational Intelligence Magazine, 55-75.

About the Authors

Seda YILDIRIM graduated from Selcuk University, Faculty of Engineering and Architecture, Computer Engineering in 2015. She received the master's degree from the Department of Computer Engineering, Institute of Science, Konya Food and Agriculture University, Konya, Turkey (2019). She continues his Ph. at Konya Technical university. She has been Lecturer at KTO Karatay University since 2020. His research interests include image processing, natural language processing, artificial intelligence, machine learning and deep learning.

E-mail: seda.yildirim@karatay.edu.tr, **ORCID:** 0000-0003-2944-6826

Similarity Index

The similarity index obtained from the plagiarism software for this book chapter is 19%.

To Cite This Chapter:

Yıldırım S. (2022). Natural Language Processing Applications in Engineering. In S. Kocer & O. Dundar (Eds .), *Current Studies in Basic Sciences Engineering and Technology* (pp. 12–30). ISRES Publishing.