# Sentiment Analysis

SOFIA UNIVERSITY
ST. KLIMENT OHRIDSKI
EST. 1888

FACULTY
OF ECONOMICS
AND BUSINESS
ADMINISTRATION

## What you will learn

- What is **Sentiment analysis**?
- Why do we need it? **Practical applications**.
- **Main methods** for sentiment analysis (including some key concepts in machine learning).
- How to **apply** them in **Python**?

I.

# But...what exactly is Sentiment Analysis?

“

***Sentiment analysis (SA)**, also called **opinion mining**, is the field of study that analyzes people’s opinions, **sentiments**, evaluations, appraisals, attitudes, and emotions towards **entities** such as products, services, organizations, individuals, issues, events, topics, and their **attributes**.*

”

***Bing Liu, 2012***

## Sentiment Analysis - history

- Exists as a concept since the beginning of the 20$^{th}$ century.
- Turns into a **hot topic** around 2002-2004 -> **99%** of studies are published **after 2004**[1].
- The **social media boom** – availability of text data expressing people's opinions, sentiments, emotions etc.
- Very important in **NLP**, **text analytics** and **social** and **management** sciences (economics, politics etc.)

---

[1] Mäntylä, Mika V., Daniel Graziotin, and Miikka Kuutila. "The evolution of sentiment analysis—A review of research topics, venues, and top cited papers." *Computer Science Review* 27 (2018): 16-32.

## Sentiment Analysis – many practical applications!

- **Find customer opinion, sentiment, pain points, needs** – products and services improvement; brand management; customer retention etc.
- Develop **emotionally intelligent** chatbots/dialogue systems.
- **Political campaigns/surveys/government issues** – for example, Donald Trump 2016 campaign Cambridge Analytica: how did it turn clicks into votes? | Big data | The Guardian
- **Trading** – detect market sentiment (in Twitter, for example).
- **And many more**…

## How do we define "an opinion" in sentiment analysis?

- Opinions consist of **5 key elements** (Bing Liu, 2012):

$$(e, a, s, h, t)$$

**Opinion Target**

**e** – target entity (for example, iPhone)

**a** – aspect of entity **e** (e.g., battery, screen etc.)

**s** – sentiment on aspect **a** of entity **e** (positive/negative/neutral)

**h** – opinion holder

**t** – opinion posting time

# Aspect-based Sentiment Analysis

$e$ $\quad a_1 \quad s_1 \quad a_{11}$

*„the iPhone camera is good, the resolution is high, however the battery life is poor“*

$s_{11} \quad a_2 \quad s_2$

Target entity – iPhone ($e$)

Aspect 1- iPhone’s camera ($a_1$)

Attribute of Aspect 1 - iPhone’s camera resolution ($a_{11}$)

Aspect 2- iPhone’s battery ($a_2$)

Sentiment – Aspect 1 – positive ($s_1$)

Sentiment of Aspect 1, Attribute 1 – positive ($s_{11}$)

Sentiment – Aspect 2 – negative ($s_2$)

# Named entity recognition (NER)

- NER - **locate** and **classify named entities** mentioned in unstructured text into pre-defined categories such as person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc.

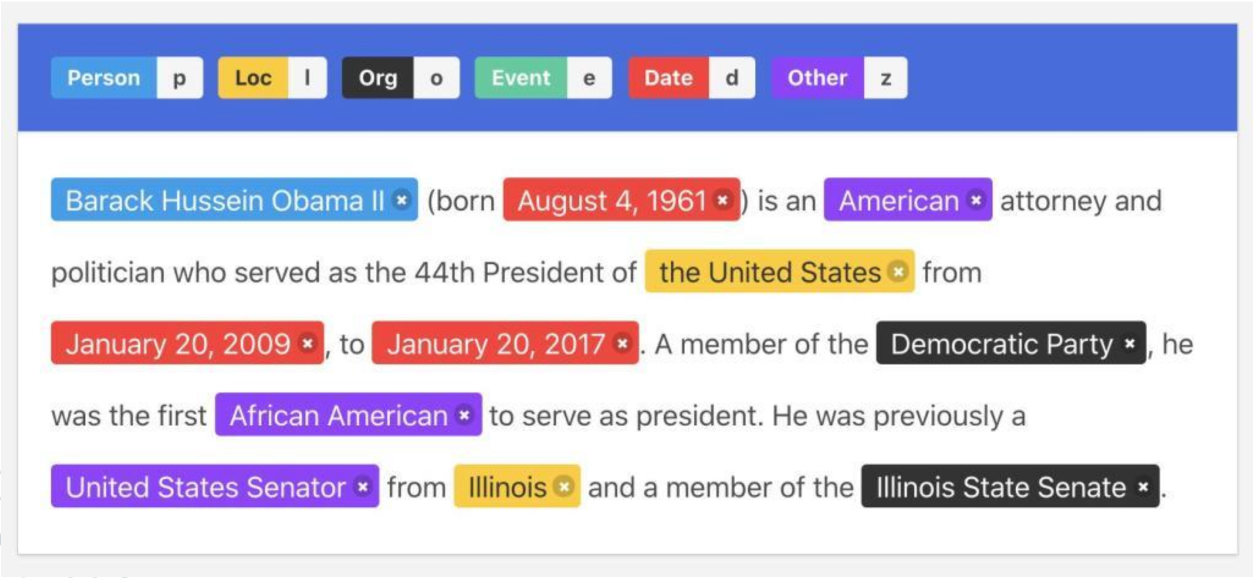





Image credits: [A Beginner's Introduction to NER (Named Entity Recognition) (analyticsvidhya.com)]

## How to simplify the task?

- In many practical applications **“entity”** ≡ **“aspect”** (the aspect is “general”).
- We know the entity in advance – e.g. scrape tweets according to keywords.
- Use **keywords** to split text data into different groups and then apply SA.
- Apply **topic modeling/clustering** and then apply SA.

## How do we define sentiments?

- Most popular task – **sentiment polarity analysis;**
- Sentiment categories **– positive, negative, (neutral);** 
- **The "Neutral" category** – optional; can imply "neutrality" or "no opinion" (only factual information);
- Sentiment categories may be also expressed as **ratings** (e.g., star ratings from 1 to 5). 

## But... what about emotion mining?

- Similar to the sentiment analysis task, but more **fine-grained.**
- An example set of emotions: 

**anger, anticipation, disgust, fear, joy, sadness, surprise, trust**

- But....What is the difference between emotion, mood and feeling?

## The most popular task

$$(\_, \text{GENERAL}, s, \_ , \_)$$

***Positive*** ***Negative***

*"Given an opinion document* **d** *evaluating an entity, determine the overall sentiment* **s** *of the opinion holder about the entity, i.e., determine* **s** *expressed on aspect GENERAL where the entity* **e***, opinion holder* **h***, and time of opinion* **t** *are assumed known or irrelevant (do not care). "*

*Bing Liu, 2012*

## Subjectivity Analysis

- Subjectivity analysis -> **Distinguish** sentences (called objective sentences) that express **factual information** from sentences (called subjective sentences) that express **subjective views and opinions**.
- **Be careful**! Objective sentences are not always neutral and can imply sentiments (also known as fact-implied opinions):

*“My new iPhone broke on the second day.”*

- Another example – news expressing **desirable/undesirable facts**.

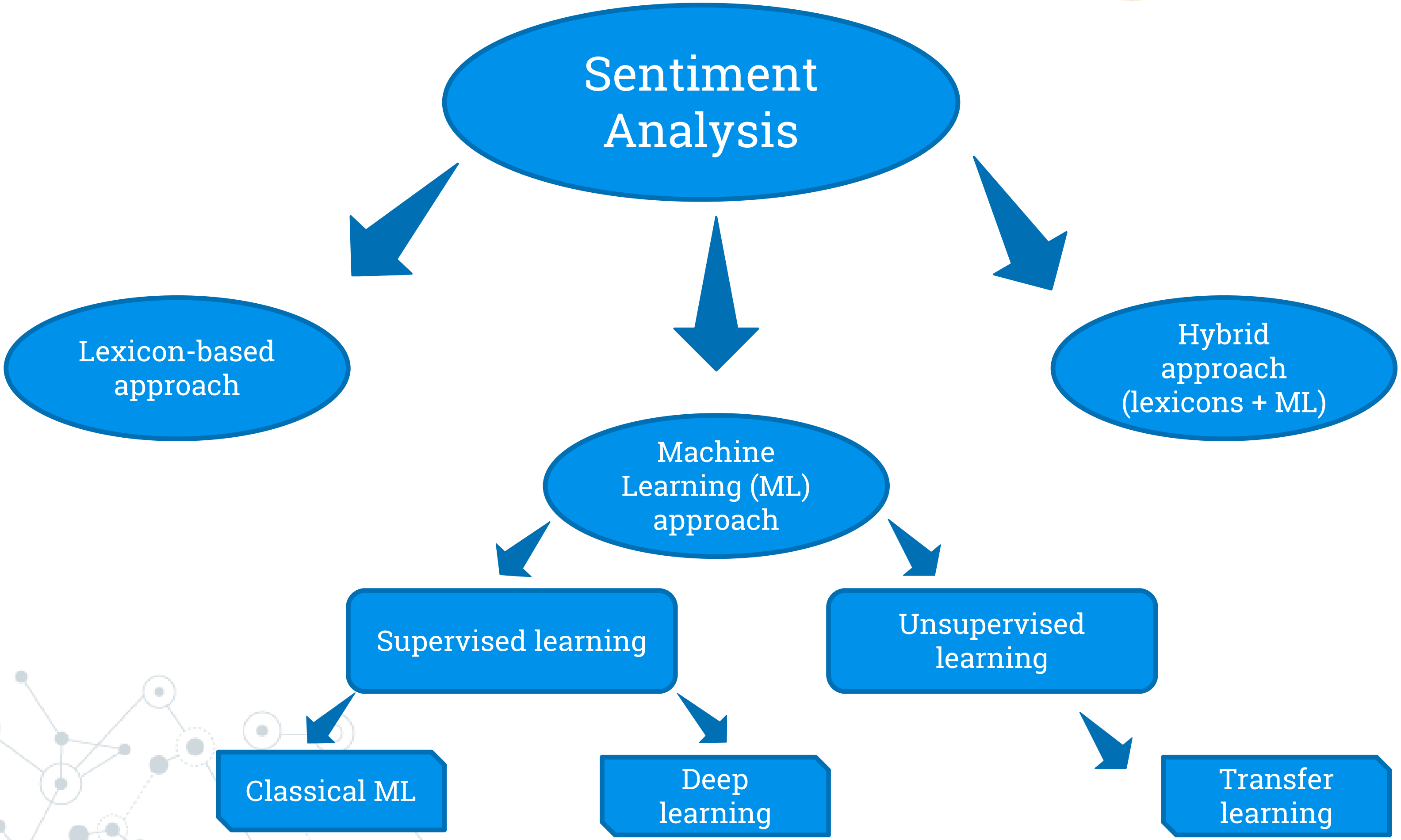
## Levels of sentiment analysis

Sentiment analysis can be applied on **three different levels:**

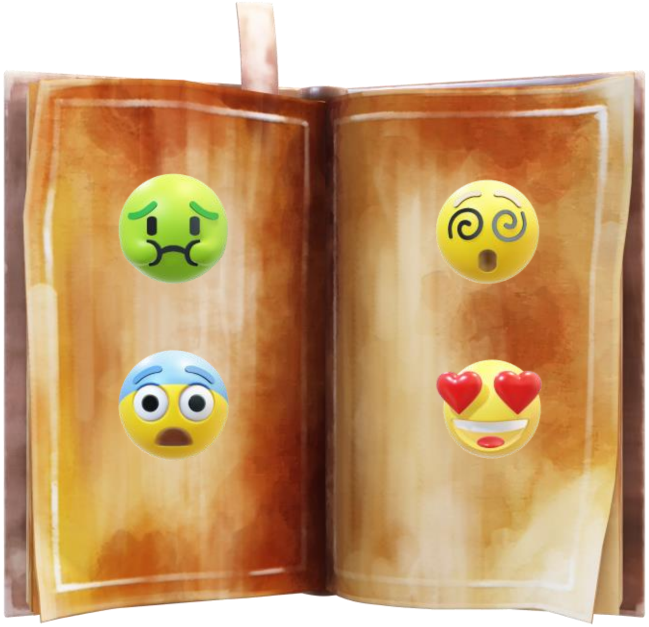
1. Document level
2. Sentence level
3. Aspect level

Difficulty level rises

# Main methods for sentiment analysis



# II. Sentiment lexicons



## Wait, what?

- A **sentiment lexicon (dictionary)** is a collection of words (also known as polar or opinion words) associated with their **sentiment orientation** (positive or negative).
- Example of **sentiment words** - good, wonderful, bad, amazing etc.
- Sentiment words can be mapped to **categories/numerical ratings**.

## Methods for sentiment lexicon generation

- **Manual** approach – label words manually (very time consuming).
- **Dictionary-based** approach – use “seed” words and then expand the list with the help of a large dictionary (synonyms/antonyms). For example - WordNet | A Lexical Database for English (princeton.edu)
- **Corpus-based** approach – takes into account the data domain (for example, “freeze” can be both positive or negative depending on the context). This approach is based on statistical learning applied on labeled text data corpus in a chosen domain (for example, social networks).

## Available Sentiment lexicons

- SentiWordNet - NLTK :: Sample usage for sentiwordnet
- AFINN - afinn · PyPI
- VADER - https://github.com/cjhutto/vaderSentiment
- SocialSent - SocialSent: Domain-Specific Sentiment Lexicons (stanford.edu)
- TextBlob - https://textblob.readthedocs.io/en/dev/quickstart.html
- NRC emotion lexicon - NRCLex · PyPI (includes sentiments and emotions)
- Other options are also available – in R, Azure, online tools etc.

## Pros and Cons of sentiment lexicons

**Cons:**

- General purpose lexicons **do not take into account** domain specific meanings of words.
- A text might contain a sentiment word without expressing an opinion (for example, “suggest me a good movie”).
- The lexicon might contain only a few words from your text data sample.

**Pros:**

- Fast, easy to understand and apply, no need for training data!

# III. Supervised ML

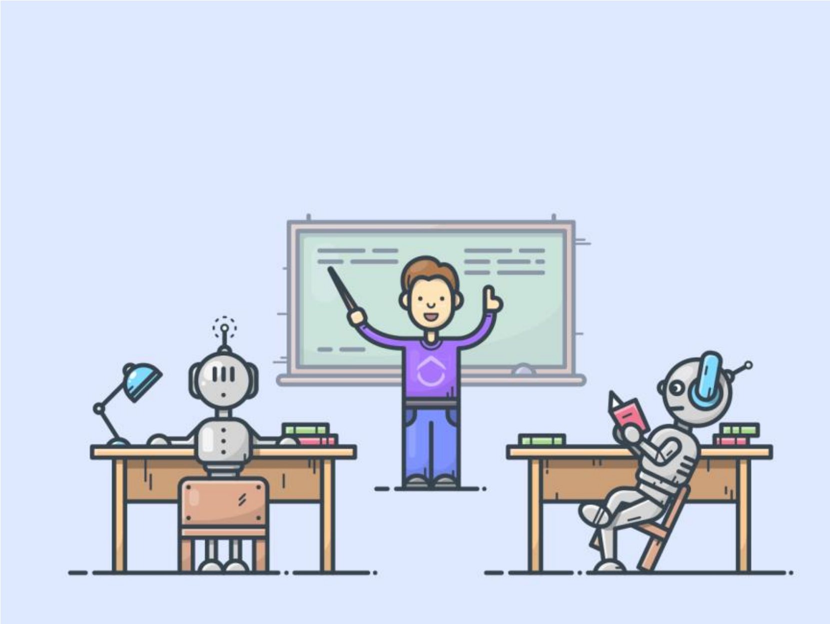


Image credits: [Supervised Vs. Unsupervised Machine Learning | ClickUp Blog]

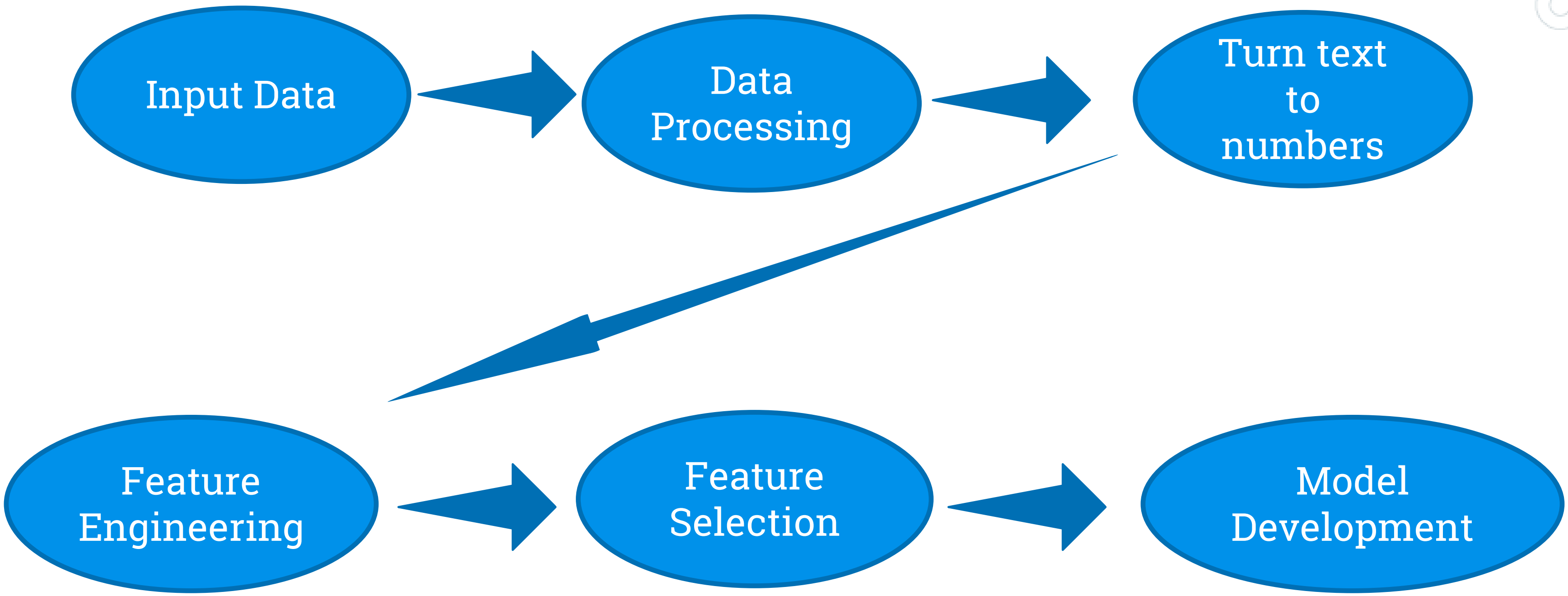
## Supervised vs. Unsupervised learning

- **Supervised learning** – predict the value of the target variable based on historical data. You have training (labeled) data!
- **Unsupervised learning** – You don't have any previous knowledge about target variable values (no training data).

## Sentiment analysis as a text classification task

- “Predict the sentiment category of a given text” – this is a **text classification task.**
- **Classification** vs. **Regression** in ML?
- Text classification **examples** – spam vs. not spam; positive vs. negative; fake news vs. real news etc.

# The classical ML approach for text classification



## The classical ML approach – potential sources of explanatory variables

- **Words** – unigrams, bigrams etc. (n-grams)
- **Emoticons -** ☹️ 🙂 🙄
- **Punctuation** – “!!!!!”, “?”, “?!” etc.
- **Word capitalization –** “COOL”
- **Available metadata**
- And other…

# The deep learning approach

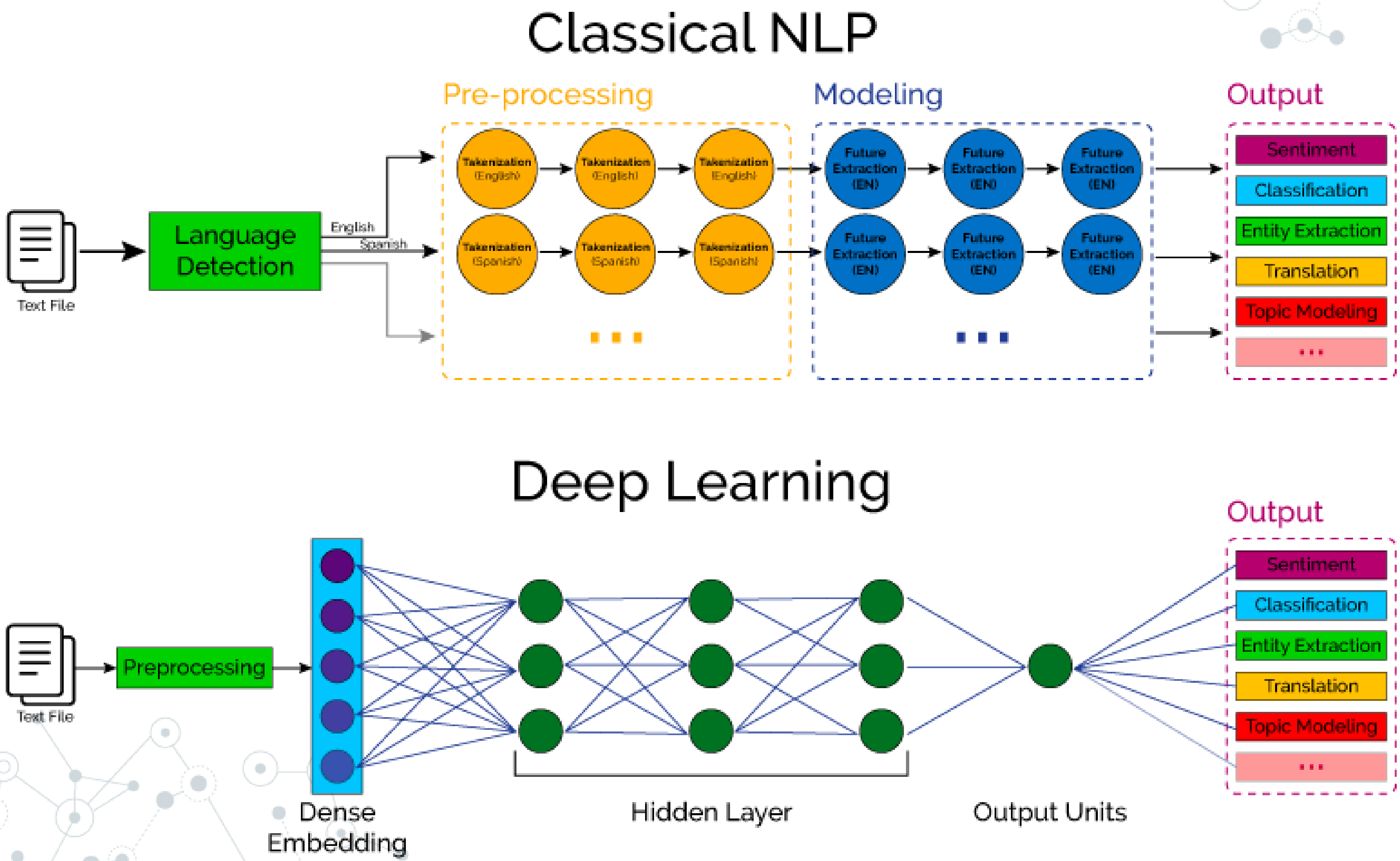



Image credits: [difference-between-classical-nlp-deep-learning-nlp – nsuchaud – insights that matters]()

## Classical vs. deep learning approach – pros and cons (1)

- Supervised ML **requires labeled data**! However, models trained with the help of supervised ML are domain-specific! (how to label data? - Amazon Mechanical Turk (mturk.com))
- The classical approach requires a lot of efforts in feature engineering and selection, unlike the deep learning approach.
- The classical approach does not scale well as data volumes increase.

## Classical vs. deep learning approach – pros and cons (2)

- If you have big data – try deep learning; If you have small to moderate data sample – try the classical approach.
- The classical approach is simpler and easier to understand.
- If you have to explain your results – use the classical approach.
- Training a deep learning model might require a lot of time and technical resources!

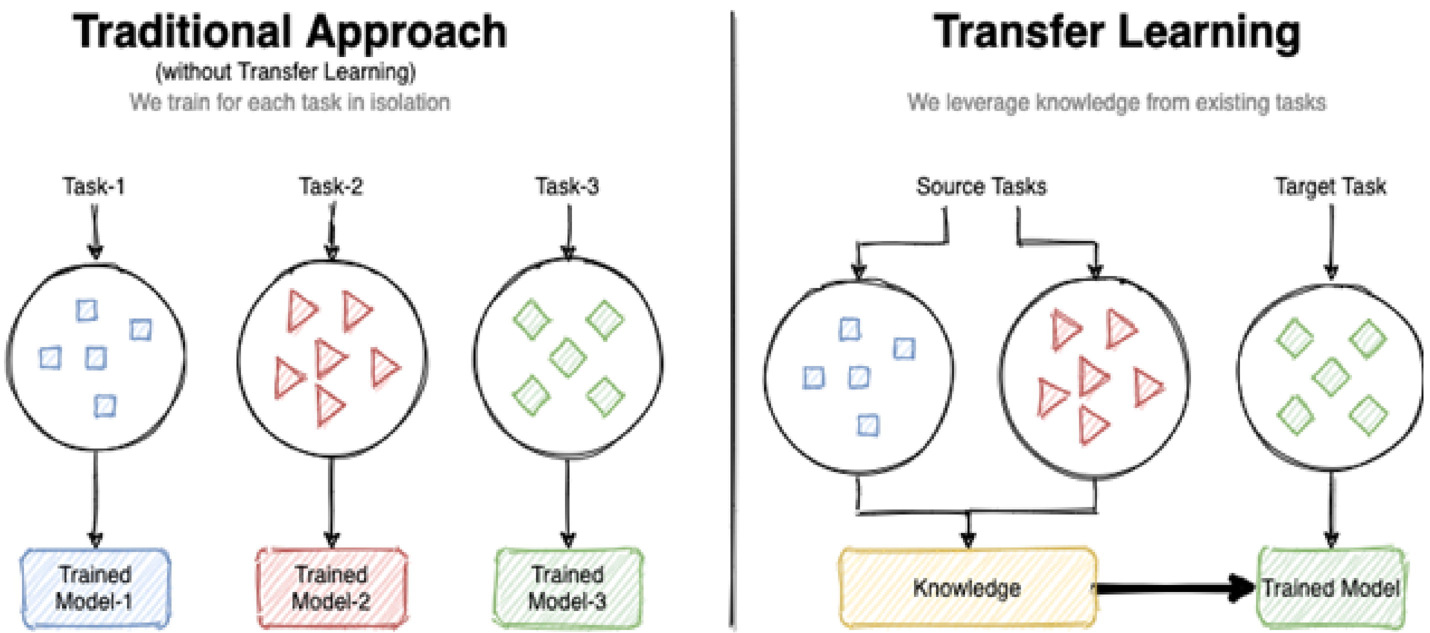
# IV. Unsupervised ML

The hype around...Transfer learning

## But first, what is transfer learning?

- Main concept – machines applying the "**human way of learning**"
- *"**Transfer learning** (TL) is a research problem in machine learning (ML) that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem."*
- *"After supervised learning, transfer learning will be the **next driver of ML commercial success**." – Andrew Ng*

# High-level overview of the transfer learning approach




Source: [1 Fundamentals of transfer learning - Transfer Learning in Action MEAP V01 (manning.com)]

## When to apply transfer learning?

- No training data.
- Small sample of training data.
- Unbalanced target distribution.

The answer: **Use <u>pre-trained models in combination</u> with transfer learning.**

## Different scenarios for transfer learning

Examples for the sentiment analysis (SA) task:

- Use a SA model trained on **product reviews in one language** and use transfer learning to predict the sentiment of **product reviews in another language**.
- Use a SA model trained on **product reviews in English** and use transfer learning to predict the sentiment of **tweets in English**.
- Use a SA model trained on product reviews and use transfer learning to **predict the emotion** expressed in product reviews (tasks/labels are different).
- And other..

## The “NLP Sesame street”

- The introduction of **Transformer models** – Google Brain, 2017.
- Extremely useful in NLP and Computer Vision.
- The **rise of pretrained models** – BERT (in 2018), ERNIE, RoBERTa, ALBERT etc.
- Learn more about **BERT** - https://huggingface.co/blog/bert-101
- The Hugging face library in Python! - https://huggingface.co/
- **But why muppets**? - https://www.theverge.com/2019/12/11/20993407/ai-language-models-muppets-sesame-street-muppetware-elmo-bert-ernie

## If you want to learn more... 🤓

- Cambria, E. et al. "A practical guide to sentiment analysis.“- **Chapter 2** (find the textbook in moodle)
- https://livebook.manning.com/book/transfer-learning-in-action/chapter-1/v-1/ - D. Sarkar and R. Bali, “Transfer Learning in Action”, **Chapter 1**
- Hugging Face – The AI community building the future.

# Thanks!

## Any questions?

You can find me at:

g.hristova@feb.uni-sofia.bg