ABCDH
https://abcdh.eu/

Distant Reading in R

Machine Learning

Simone Rebora & Giovanni Pietro Vitali
simone.rebora@univr.it giovannipietrovitali@gmail.com

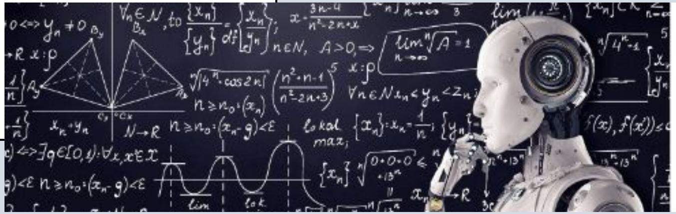
THE “TWO PARADIGMS” OF ARTIFICIAL INTELLIGENCE

Top-down

Define a set of rules which model the human cognition
Apply those rules to new subjects and situations

Dominating paradigm in the "first wave" of AI (2nd half XX century)

Main con
- absence of flexibility

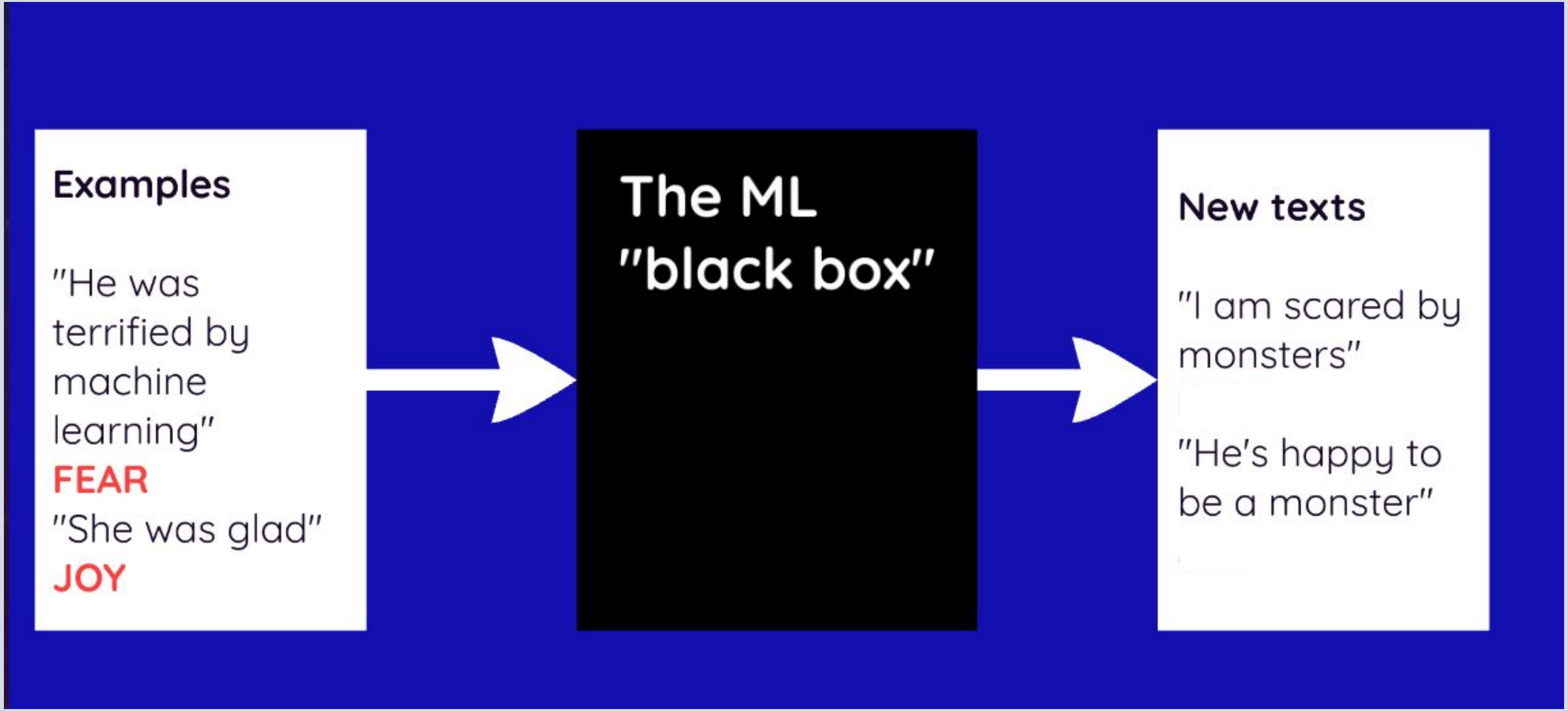


Bottom-up

Define a system that is able to "learn" a task from a set of examples (i.e. imitate the human)
Apply the system to new samples

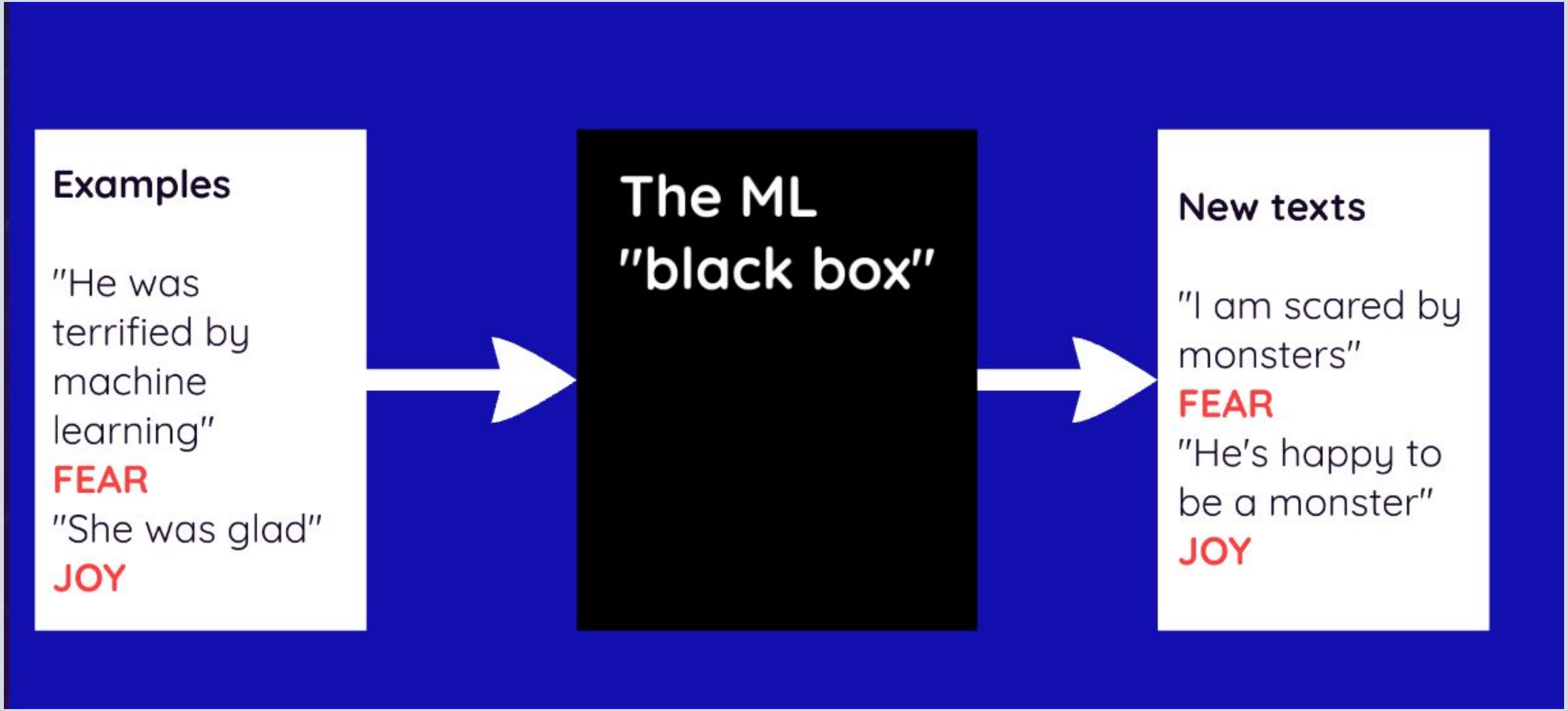
This is the machine learning approach!

ML - THE WORKING LOGIC



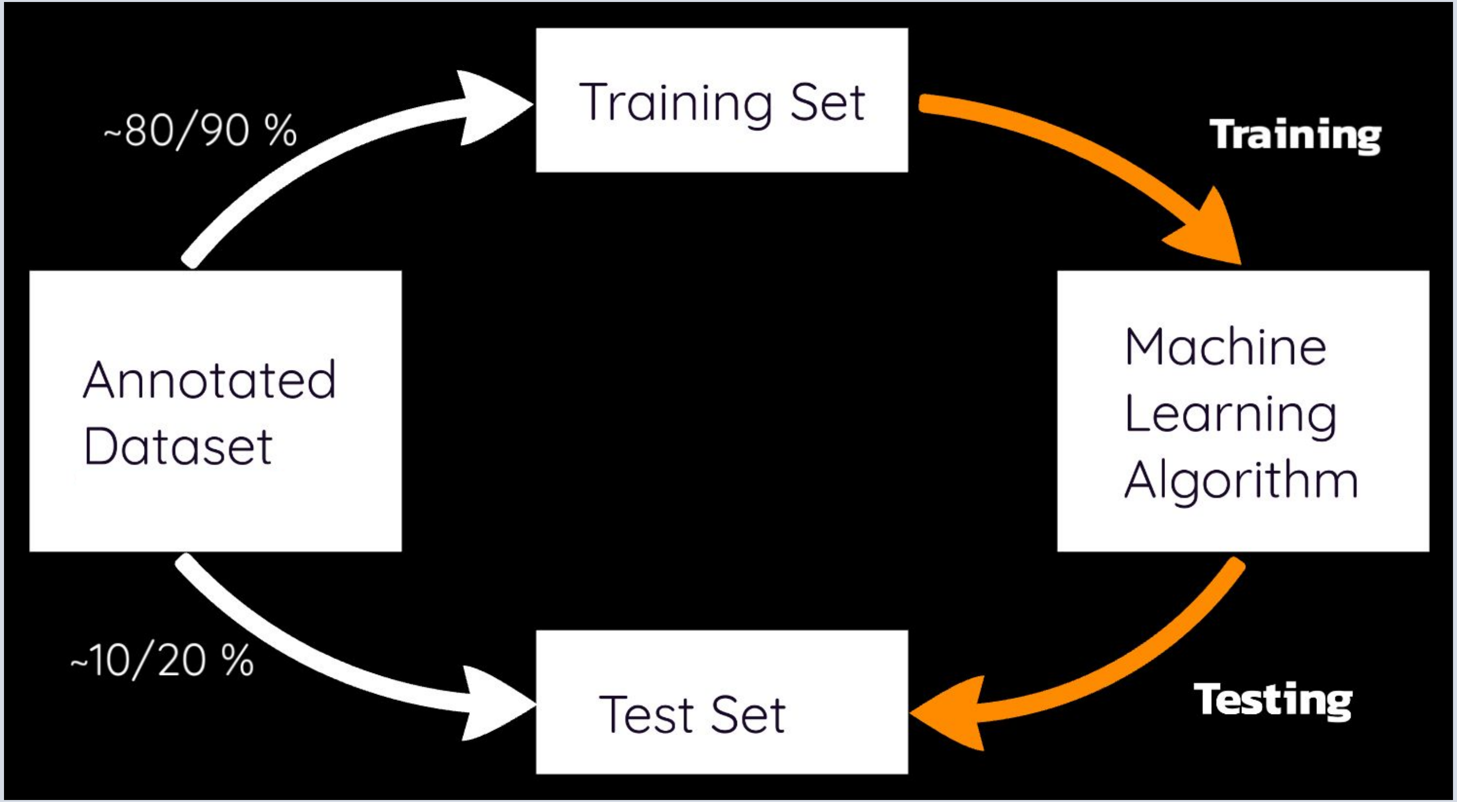
*example: sentiment analysis

ML - THE WORKING LOGIC

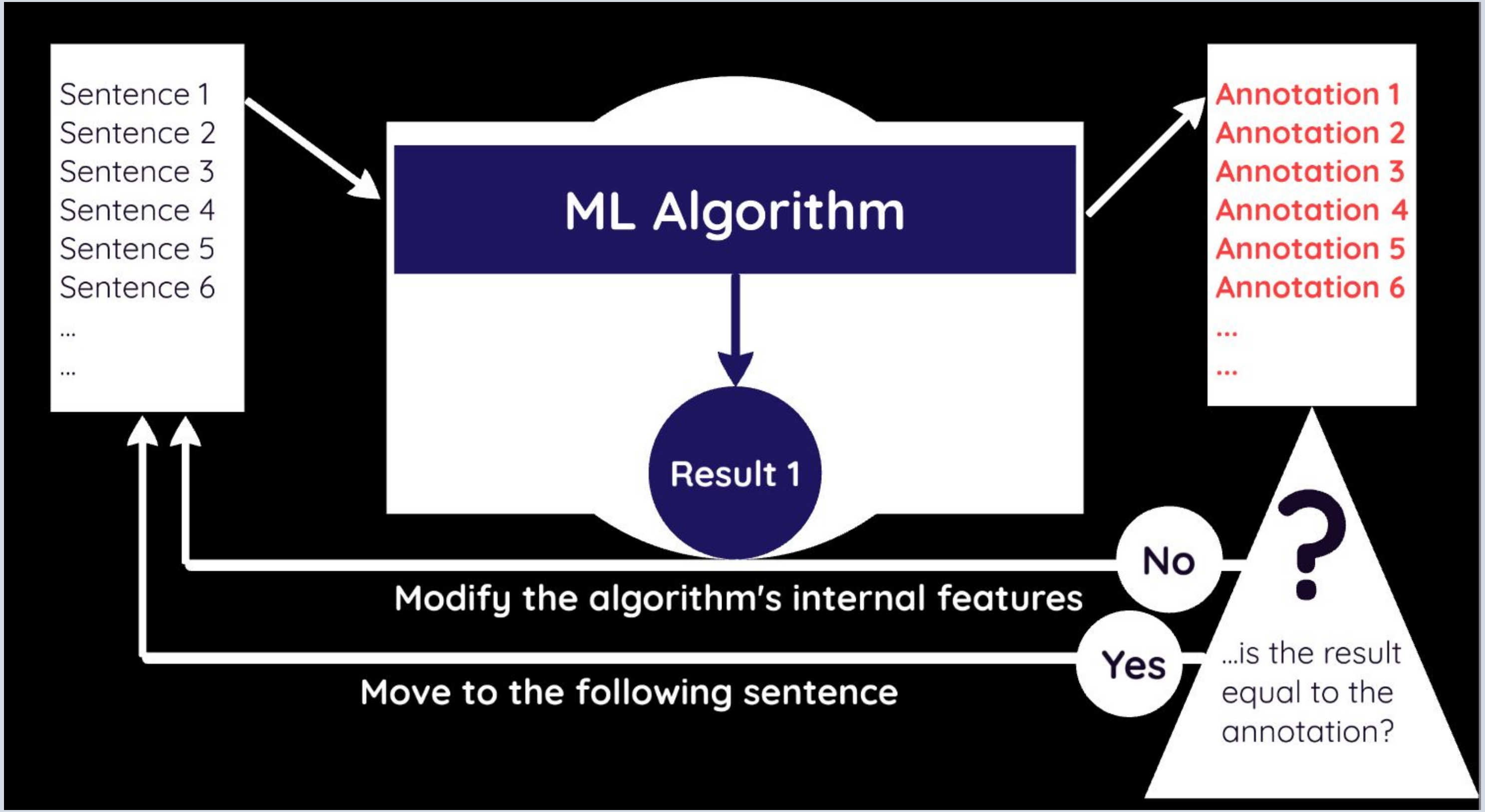


*example: sentiment analysis

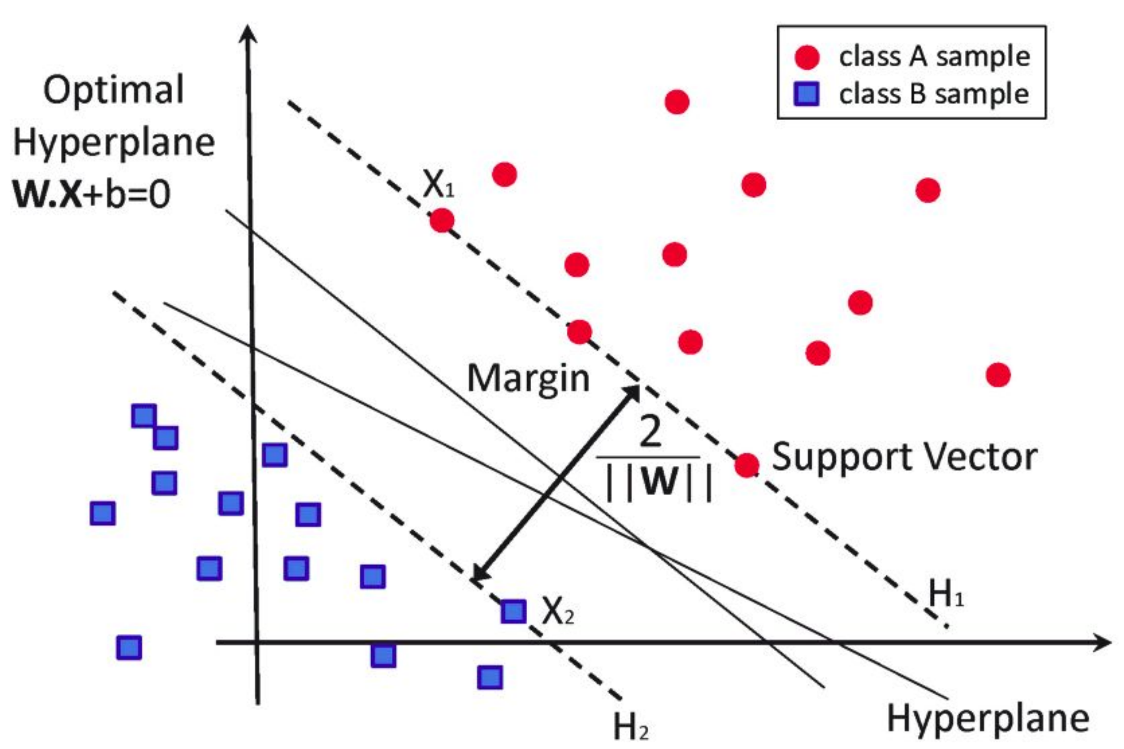
INSIDE THE ML "BLACK BOX"



TRAINING


Sentence 1
Sentence 2
Sentence 3
Sentence 4
Sentence 5
Sentence 6
...
...
ML Algorithm
Result 1
Annotation 1
Annotation 2
Annotation 3
Annotation 4
Annotation 5
Annotation 6
...
...
?
No
Modify the algorithm's internal features
Yes
Move to the following sentence
...is the result equal to the annotation?

ML ALGORITHMS



Support Vector Machines (SVM)

ML ALGORITHMS

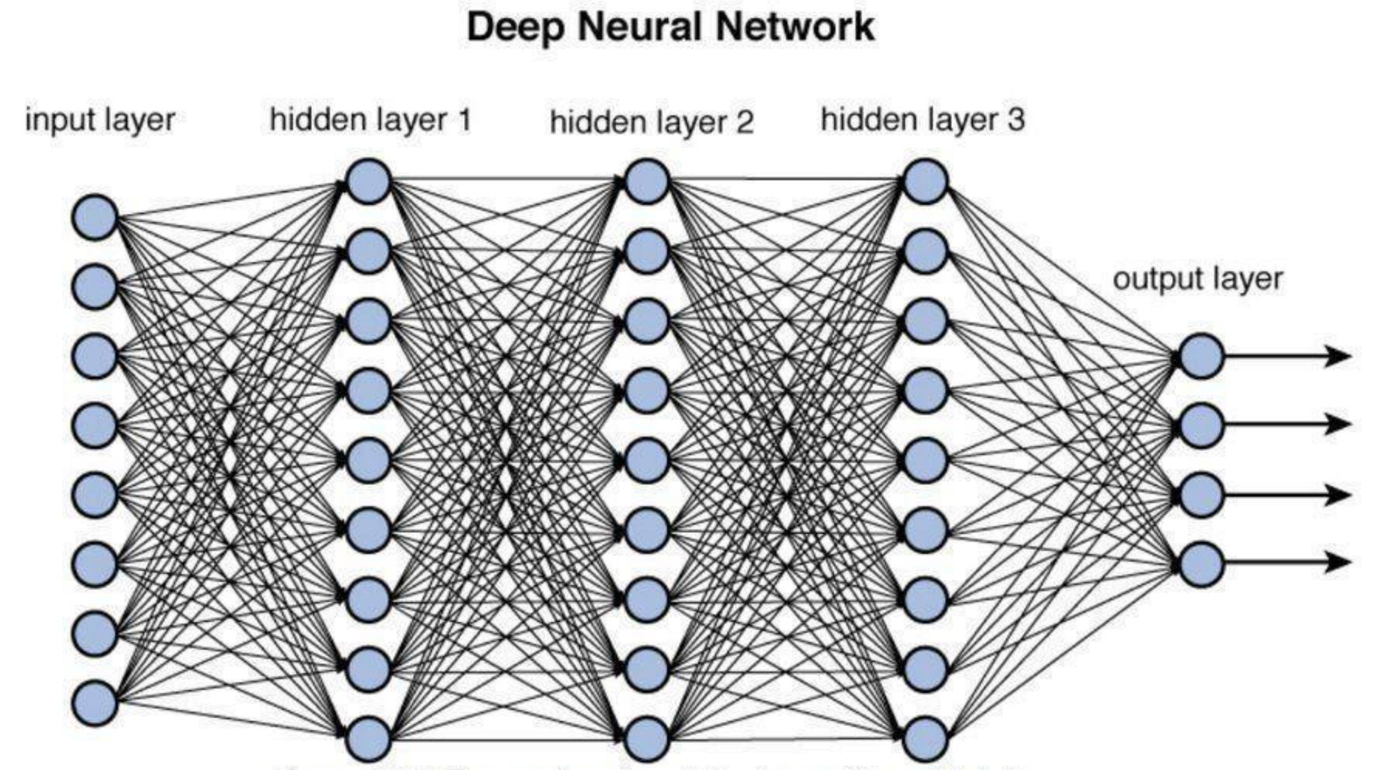
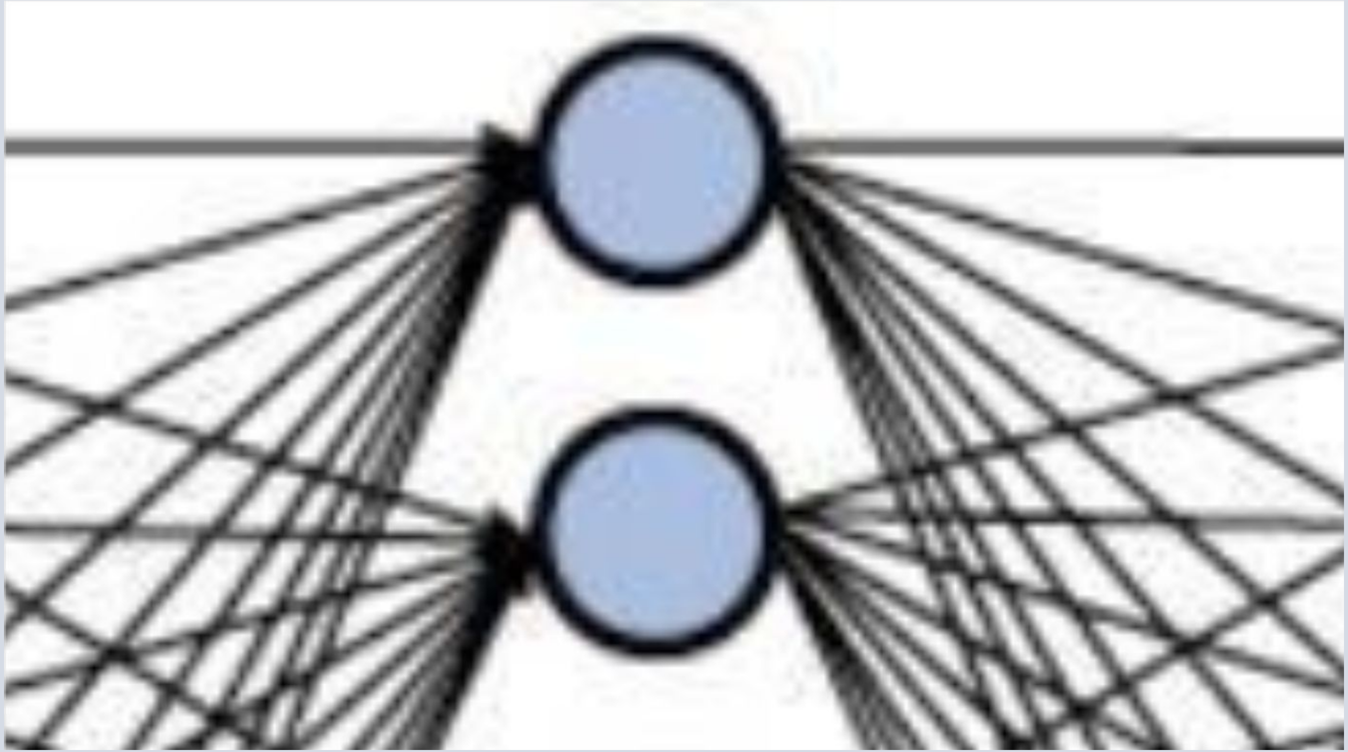
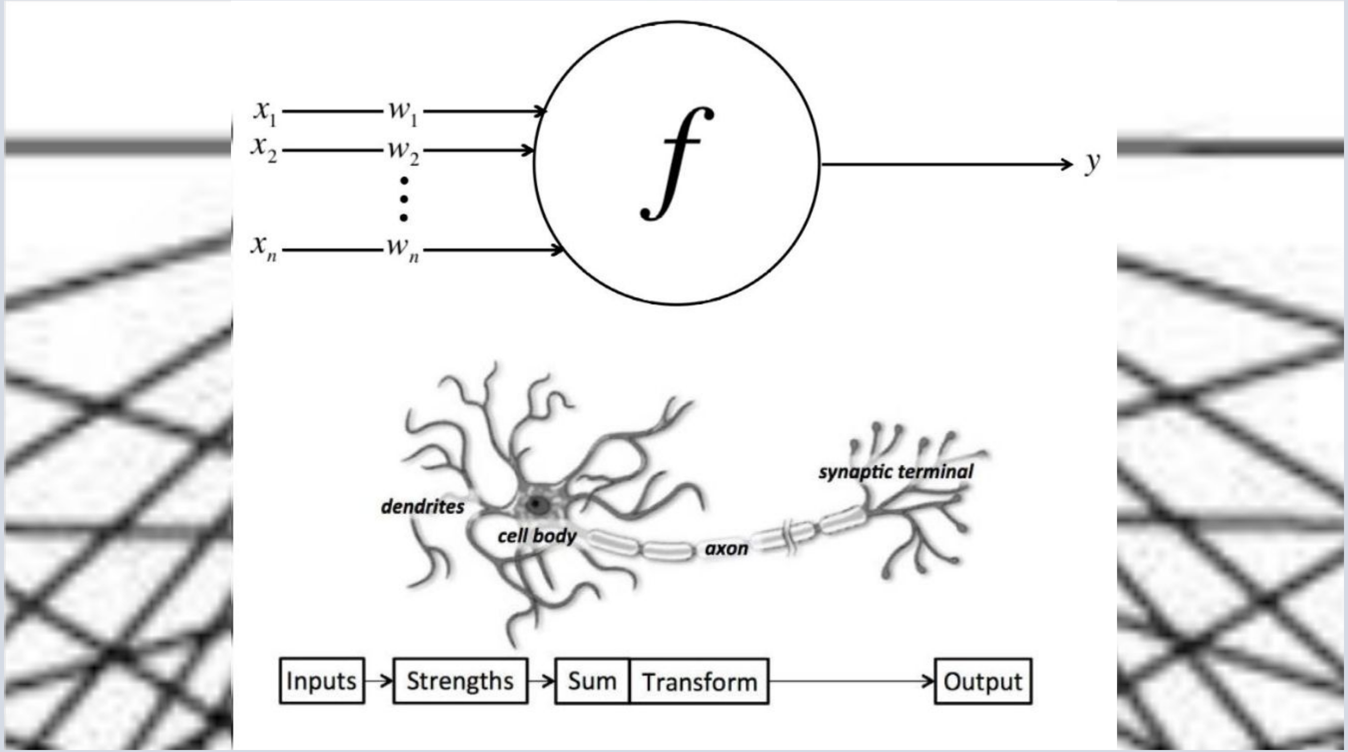


Figure 12.2 Deep network architecture with multiple layers.

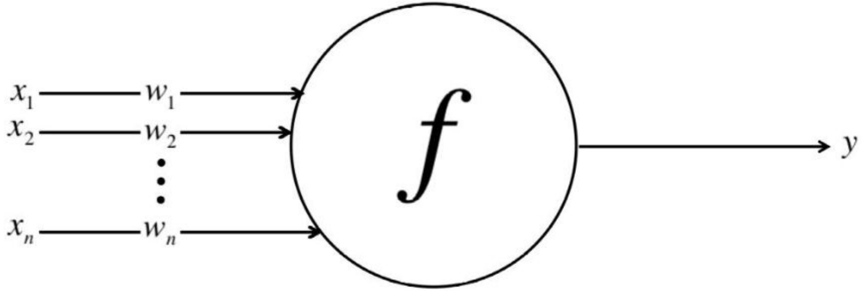
ML ALGORITHMS



ML ALGORITHMS



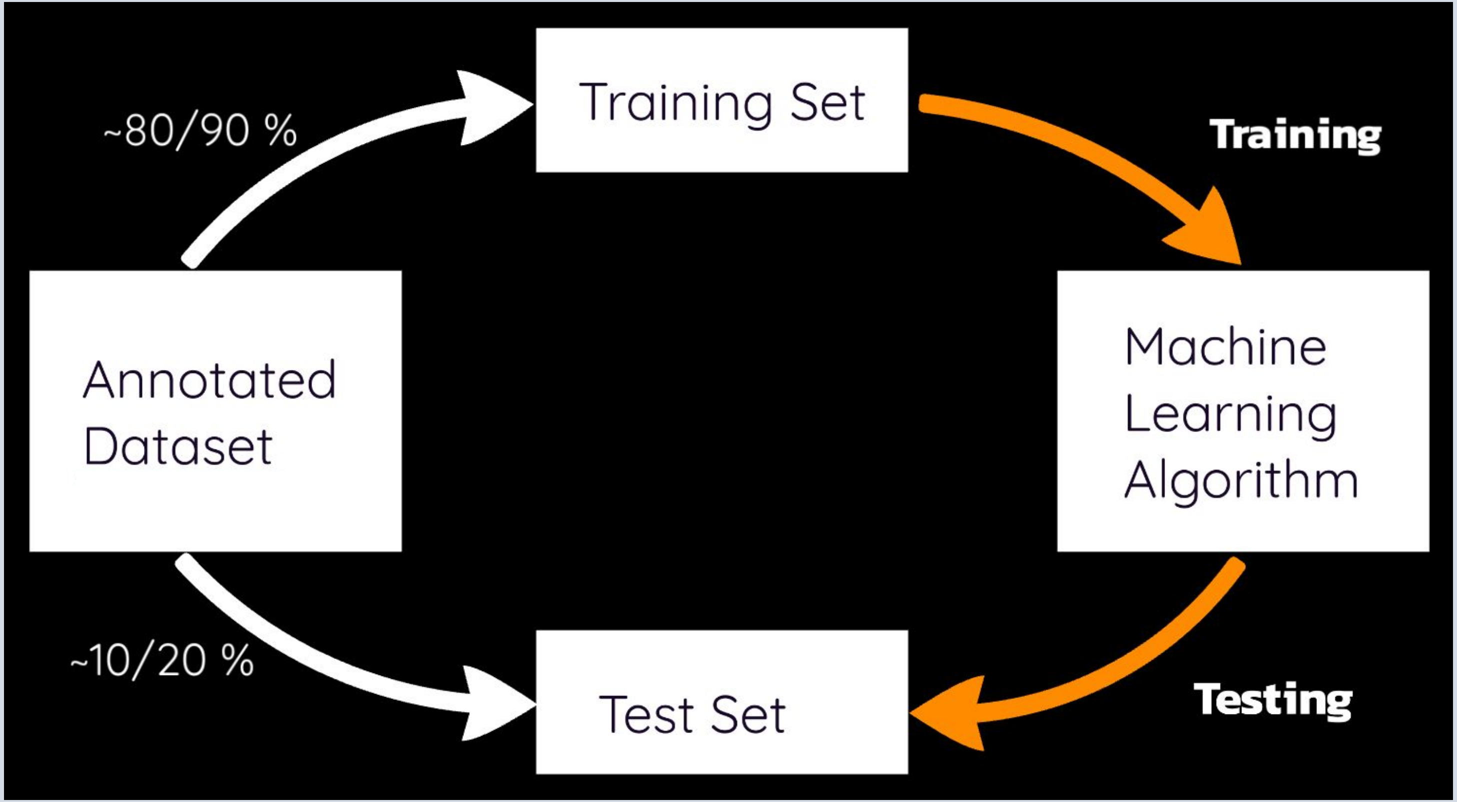
ML ALGORITHMS




```python
import numpy as np
####################################################
# Assume inputs and weights are 1-dimensional numpy #
# arrays and bias is a number #
####################################################
class Neuron:
    def __init__(self, weights, bias, function):
        self.weights = weights
        self.bias = bias
        self.function = function

def forward(self, inputs):
    logit = np.dot(inputs, self.weights) + self.bias
    output = self.function(logit)
    return output
```

INSIDE THE ML "BLACK BOX"

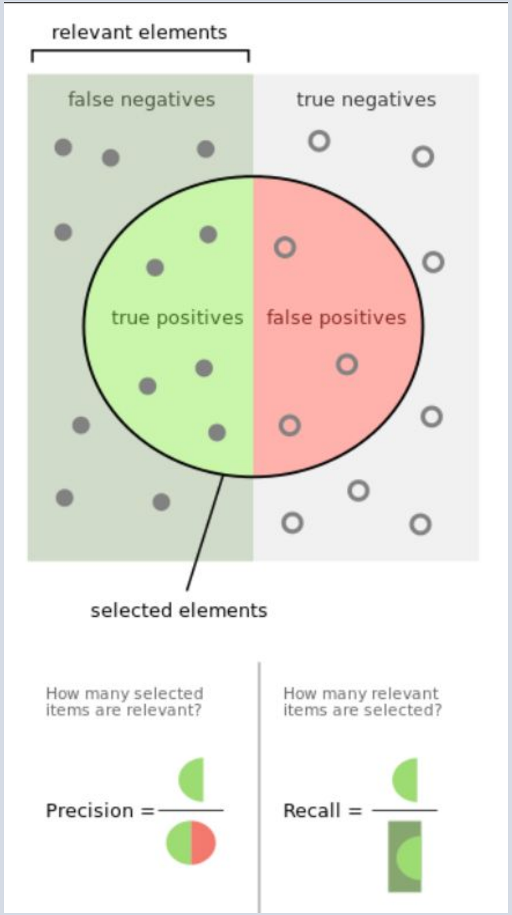


TESTING

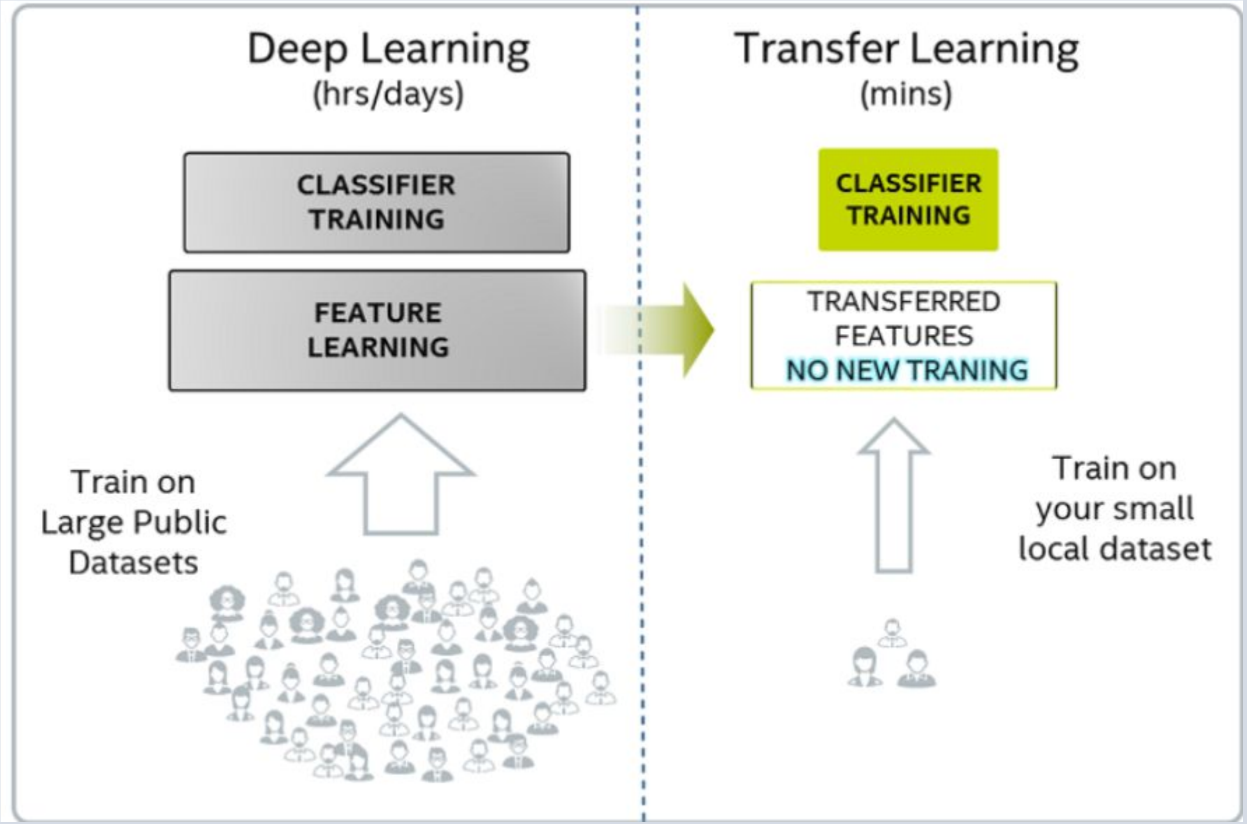
Once the training is complete, the trained algorithm is "tested" on never-seen documents

Main reason: the algorithm might have learned how to work just on the training data, but not on the task in general (overfitting)

When the task is that of assigning a label (e.g. an emotion), precision/recall are generally used



TRANSFER LEARNING



...see BERT

EMBEDDINGS

~~To~~
~~be~~
~~or~~
~~not~~
~~to~~
~~be~~

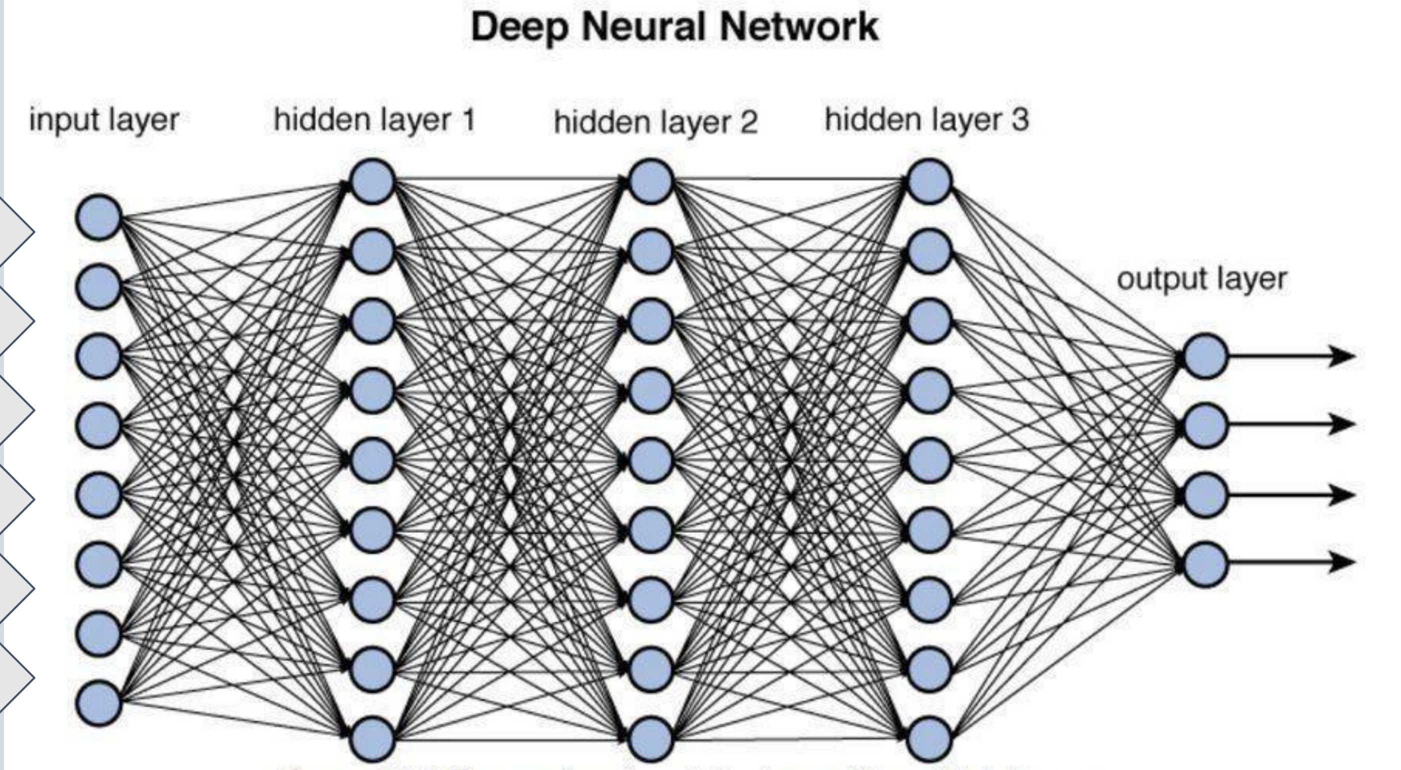


Figure 12.2 Deep network architecture with multiple layers.

EMBEDDINGS

0.123

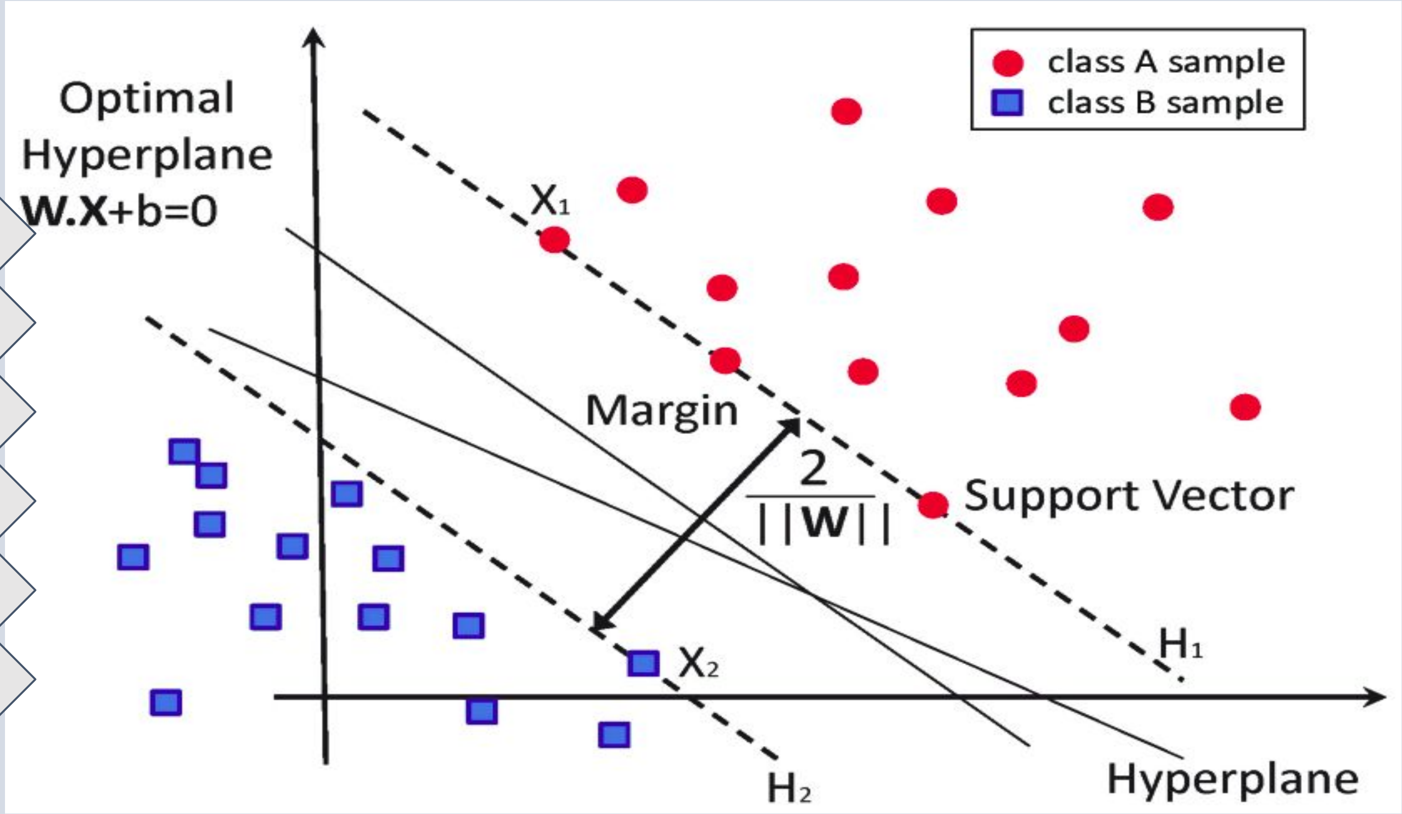
1.743

0.325

1.143

0.463

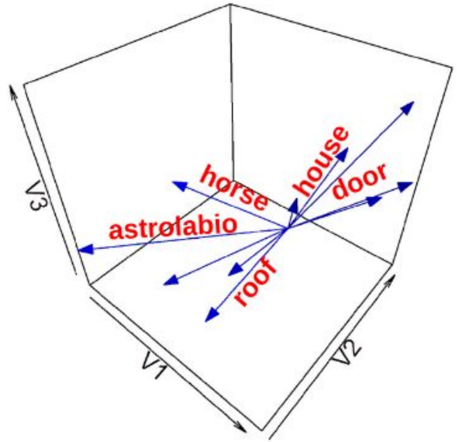
1.153



EMBEDDINGS

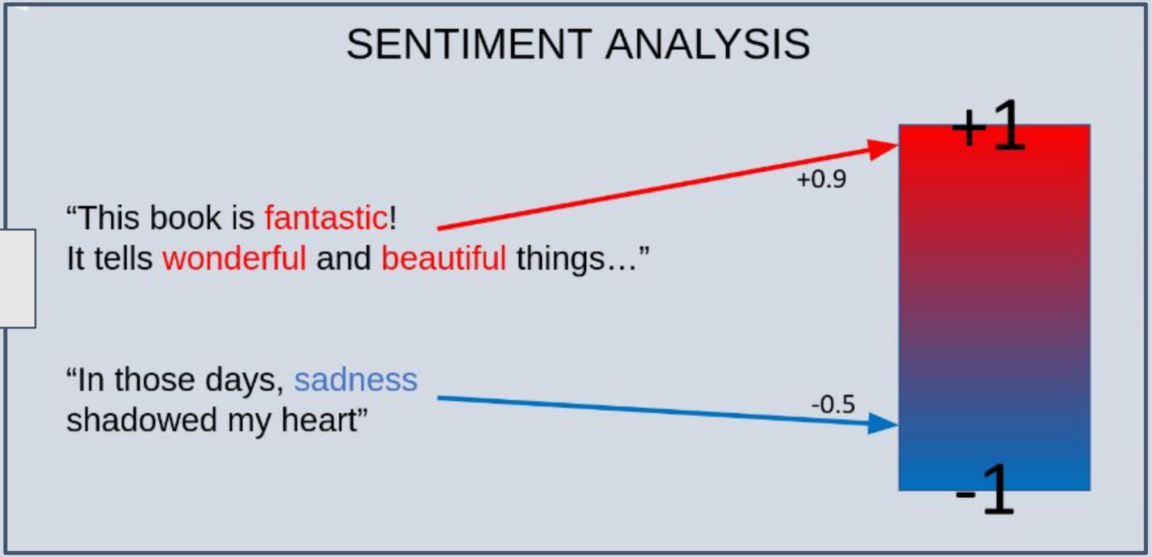
To	0.123
be	1.743
or	0.325
not	1.143
to	0.463
be	1.153

word embeddings



EMBEDDINGS

To	0.123
be	1.743
or	0.325
not	1.143
to	0.463
be	1.153



COMPUTATIONAL MODELING

"1. A model is a model *of something.* A model is always a kind of mapping. It represents something, an object, a concept, and so on, by representing it using something else like clay, words, images, and so forth.
2. A model is *not the original* and it is not a *copy of the original*. Unlike a copy, a model doesn't capture all features of the entity it represents, only some of them. The choice of features selected to be present in the model is usually based on assumptions by the creator of the model concerning which features are relevant for the intended use of the model."
(Jannidis and Flanders, 2019)

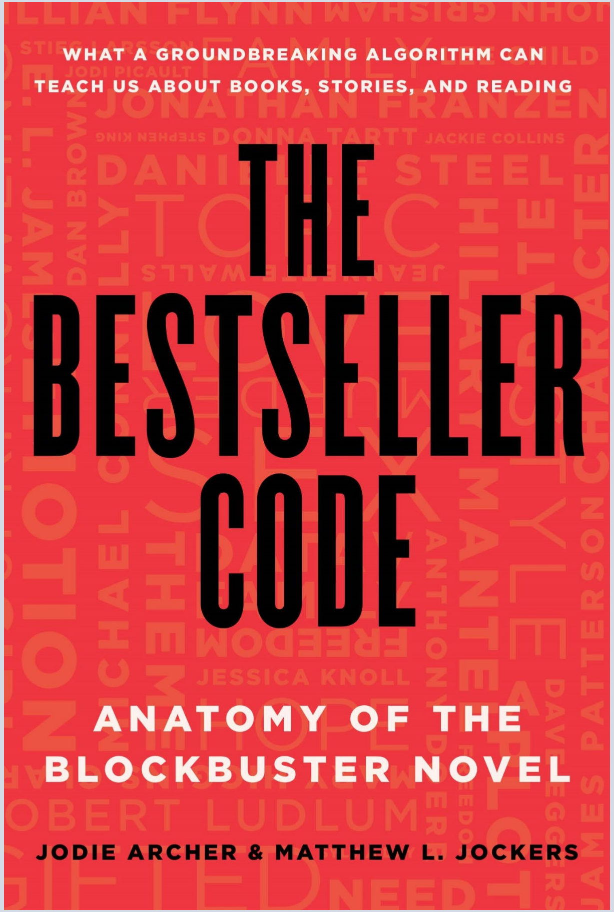
COMPUTATIONAL MODELINGS

Embedding is modeling
...because it reduces a text to a series of numbers, which represent specific features of that text

Stylometry models style
...as similarities in patterns of use of most frequent words

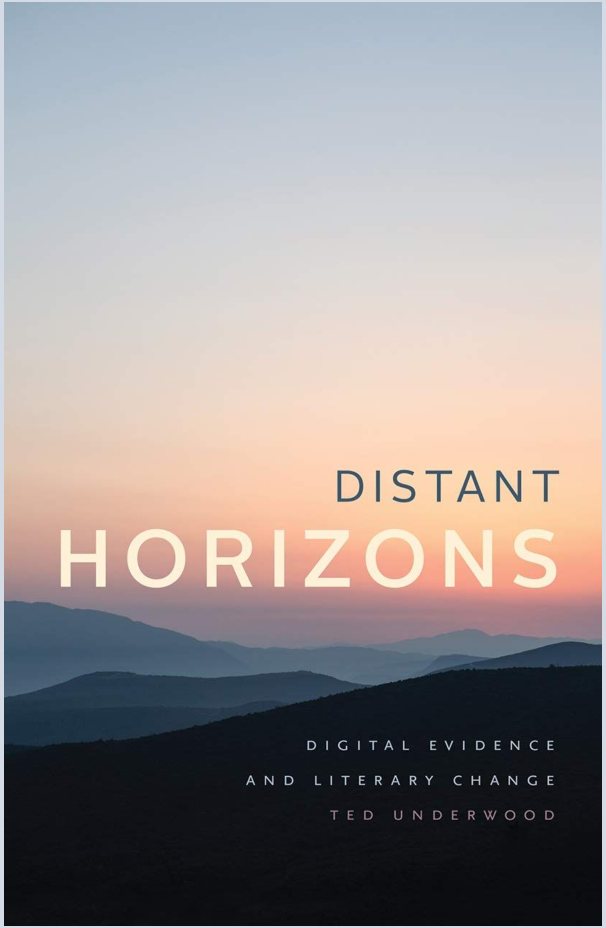
Topic models model topics
...as word probabilities based on word co-occurrences

etc. etc.



APPLICATIONS

- using a combination of sentiment analysis, topic modeling, et al. (i.e. creating embeddings)
- to train a ML classifier that predicts the commercial success of novels



APPLICATIONS

- using a simple ML classifier (logistic regression) to predict various phenomena (like genre, literariness, etc.)
- and then looking at the features that made the classifier successful (e.g. the presence of certain words, etc.)

CRITICAL ASPECTS

Opaqueness of the most advanced ML algorithms

...so you just get the results, but cannot interpret them

In ML, intelligence is intended as a form of imitation

...so can it really make predictions?

...so how can it be “creative”?

ML (and in particular transfer learning) depends heavily on the quality of the training materials

...ML models built on wide (and uncontrolled) datasets can embed strong biases

HANDS ON!

bit.ly/ESU_ML