




# Automated Evaluation of Privacy Agreements

An NLP approach to “the world’s greatest collective lie”.

By Anders Jakob Sivesind

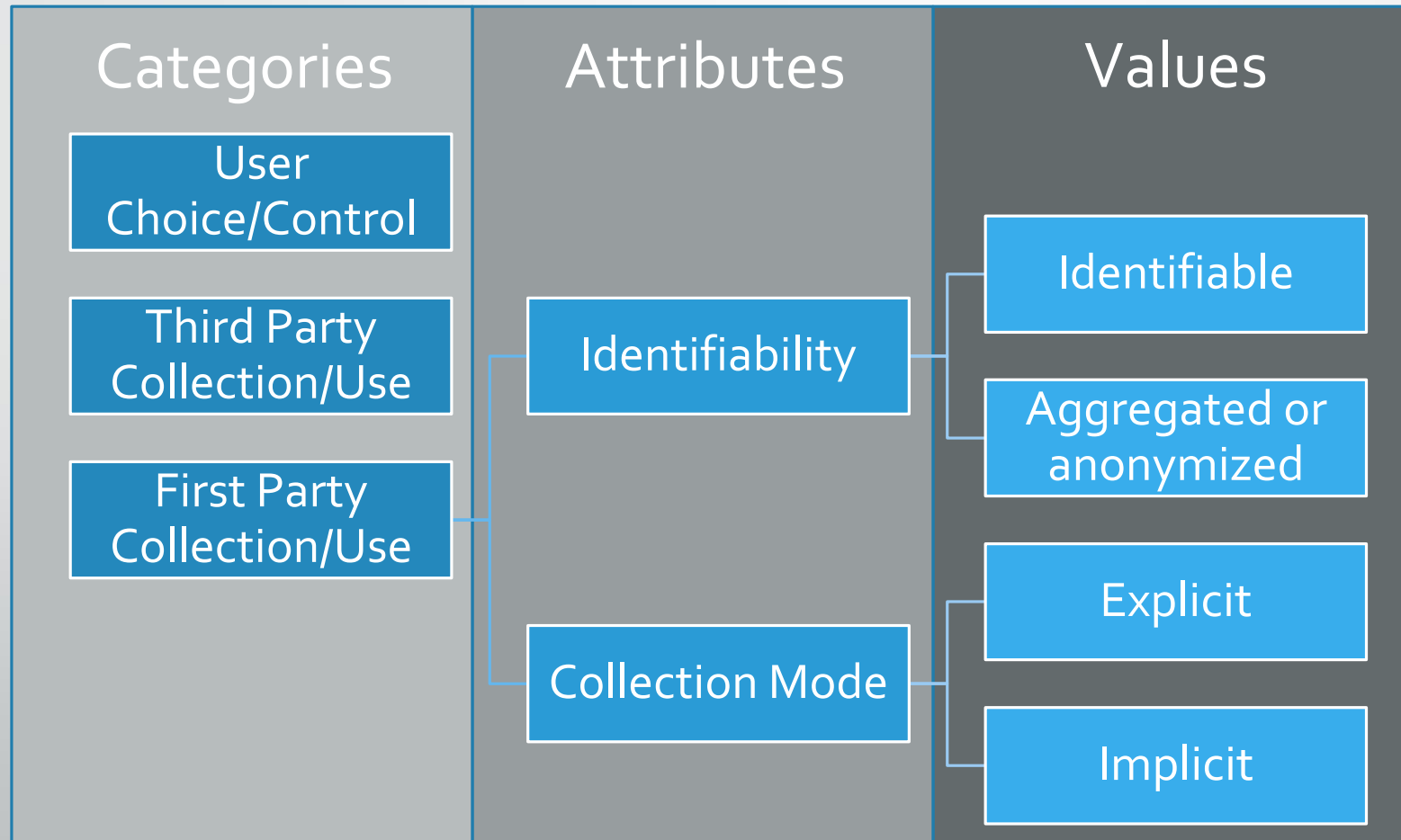
# Who am I?

- Anders Jakob Sivesind
- Masters Student
- University of Oslo
- Interested in:
  - Privacy and the inconsistency between our opinions and actions
  - Ethics and AI
  - Bottom-up AI; empower people with AI, not only companies
  - Machine Learning, in particular Graph Neural Networks



Automated extraction of privacy  
annotations from privacy agreements

# Privacy annotations



# Applications

- Use a ruleset to convert the complex privacy labels to Privacy Labels
- Provide users with labels and icons to explain the most important terms of an agreement in a few words
- Summarise privacy agreements using simple language
- Highlight parts of an agreement where the text describes a certain data practise
- Do fast, extensive surveys of current privacy agreements
- Provide preliminary checks of whether a policy follows a set of regulations

# What is a Graph Neural Network?

- A type of Neural Network
- Takes a directed graph as input
- Supports different types of edges
- Can perform:
  - Node selection
  - Node classification
  - Graph classification

# Why a GNN?

- Has not yet been used for general natural language
- Supports the use of existing tools to enrich the graph with additional information
- Performs better than other machine learning models when there is not much data available
- Text can be naturally represented as a graph

# Risks

- Traditional NLP tools struggles with the complexity of legal language. Will the graphs we extract from the text be good enough?
- Privacy policies are meant to be read by non-lawyers, perhaps the language is not as complex?
- I will be conducting experiments on corpora of varying language complexity to see how it impacts the results






# Thank you!

Any questions?

Email: [ajsivesind@gmail.com](mailto:ajsivesind@gmail.com)



“I have read and accept the  
terms & conditions.”

# Motivation

- Users should not have to blindly trust companies to give them fair agreements
- Manually evaluating privacy agreements is very slow on a big scale
- Companies could get quick preliminary checks whether their policy meets GDPR or other requirements

# Problem to solve

1. Create a data set Done: OPP-115 Corpus by Wilson et al., 2016
2. Construct an NLP algorithm that can automatically determine the content of a privacy agreement and construct Privacy Labels Work in progress
3. Use the algorithm for Future work
  1. Summarise privacy agreements for laypeople
  2. Visualise the most important content in web browsers and app stores
  3. Evaluate privacy agreements against GDPR
  4. Compare privacy agreements against requirements or regulations

# Data set

- OPP-115 Corpus, published by Wilson et al. in 2016
- Consist of 115 manually annotated Privacy Policies gathered from a variety of websites.
- Each text segment is annotated with the data practises described in the text
- Available at: <https://usableprivacy.org/data/>

# Data set

- The data practises are split into 10 **categories**, such as “First Party Collection/Use” and “User Choice/Control”

# Data set

- The data practises are split into 10 **categories**, such as “First Party Collection/Use” and “User Choice/Control”
- Each category have a set of **attributes**, which have a set of possible values.

# Data set

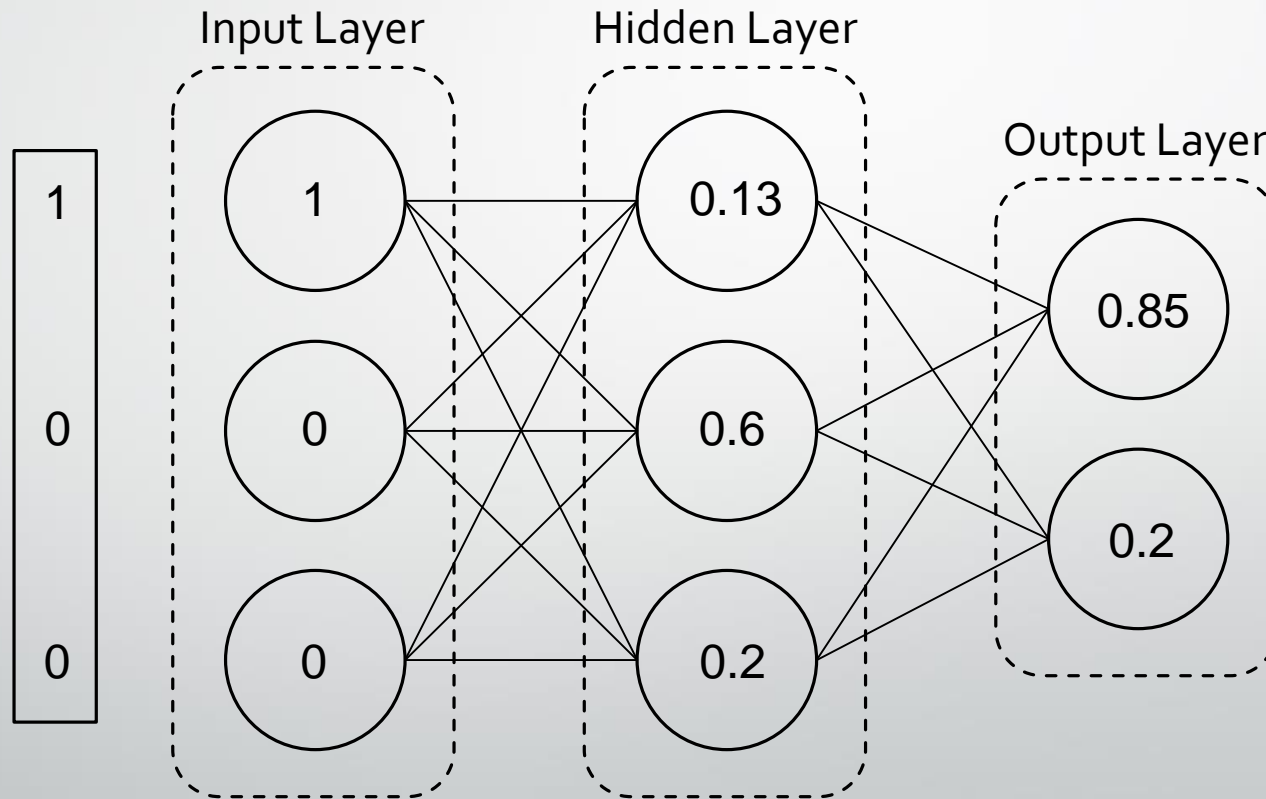
- The data practises are split into 10 **categories**, such as “First Party Collection/Use” and “User Choice/Control”
- Each category have a set of **attributes**, which have a set of possible values.
- **Example:** the category “Third Party Sharing/Collection” has an attribute “Identifiability” which may have values “Identifiable”, “Aggregated or anonymized”, etc.



# Data set

- The data practises are split into 10 **categories**, such as “First Party Collection/Use” and “User Choice/Control”
- Each category have a set of **attributes**, which have a set of possible values.
- **Example:** the category “Third Party Sharing/Collection” has an attribute “Identifiability” which may have values “Identifiable”, “Aggregated or anonymized”, etc.
- Two staged problem:
  1. Identify which category the text segment fits
  2. Determine the value of each relevant attribute for the category identified

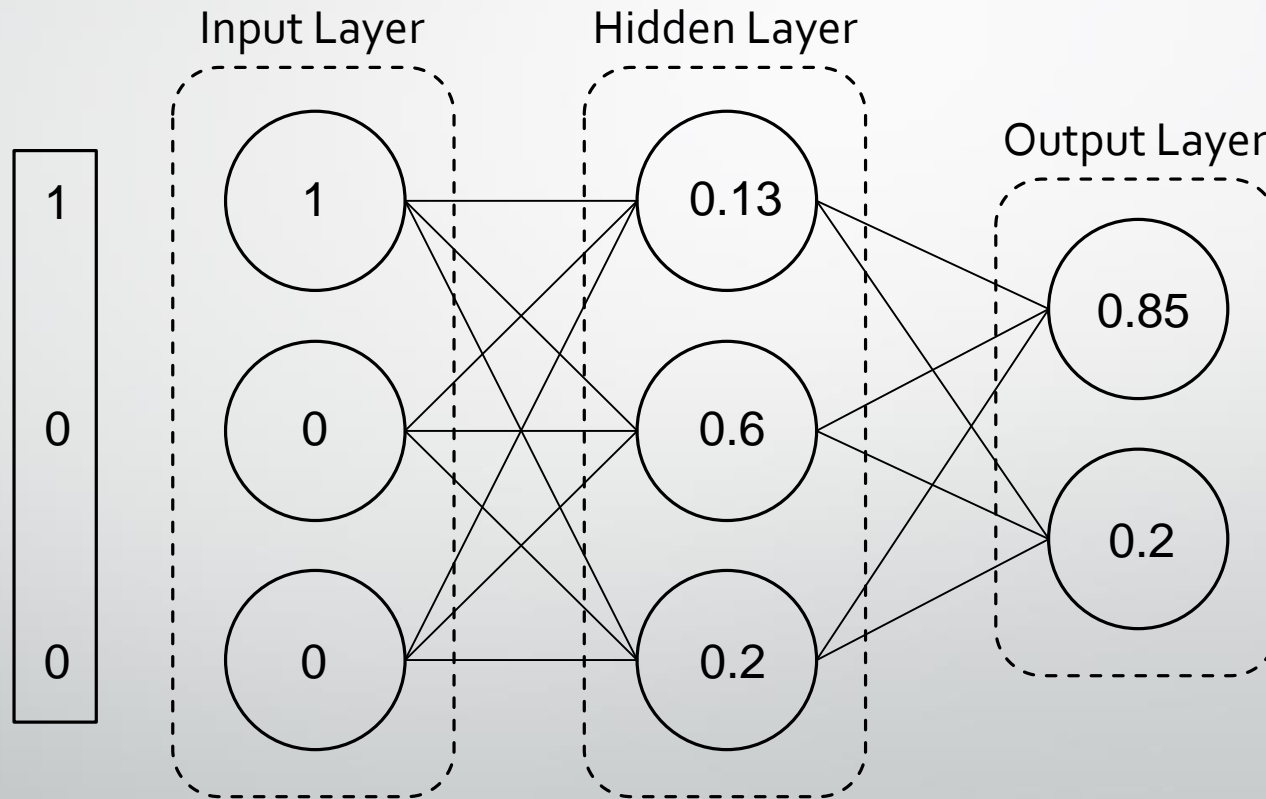
# Neural Networks



# Word Embeddings

- Vector representation of the context of a word
- Word context is a good analogy for how the word is interpreted by humans
- Example: King – Man + Woman = Queen
- Some popular Word Embedding frameworks: GloVe, FastText and Word2Vec

# Neural Networks



# Neural Networks

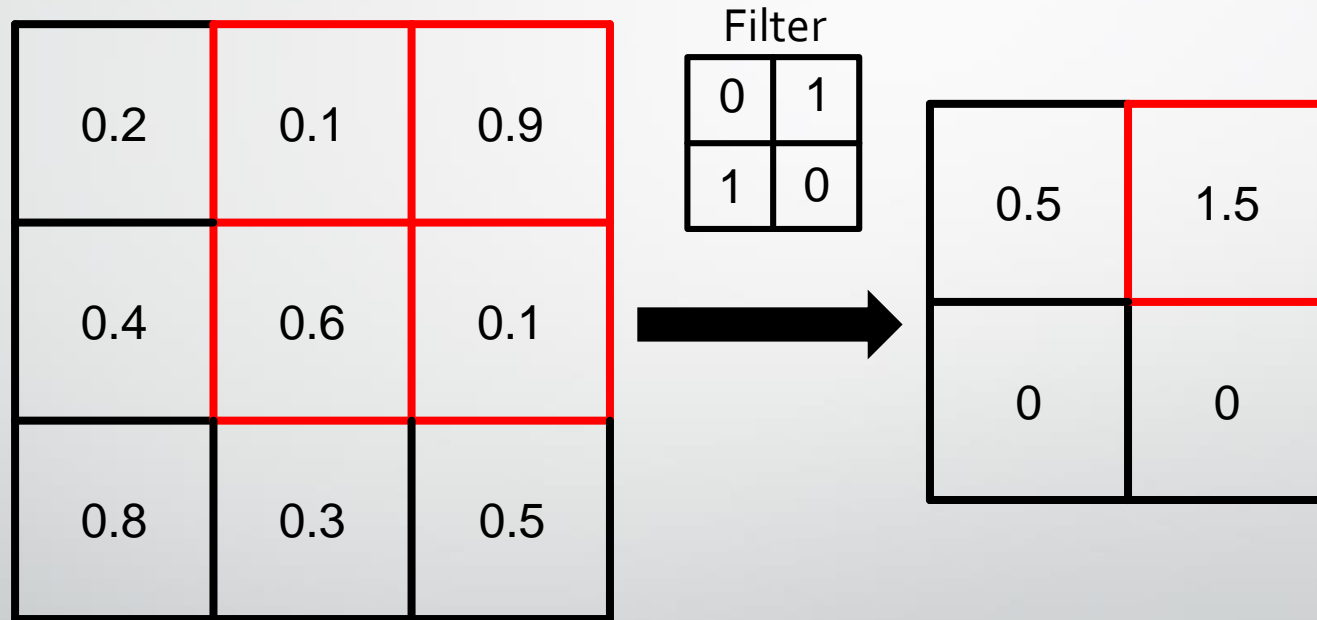
## Pros

- Learns the function from data
- Can in theory represent any function (not so in practise)
- Very versatile

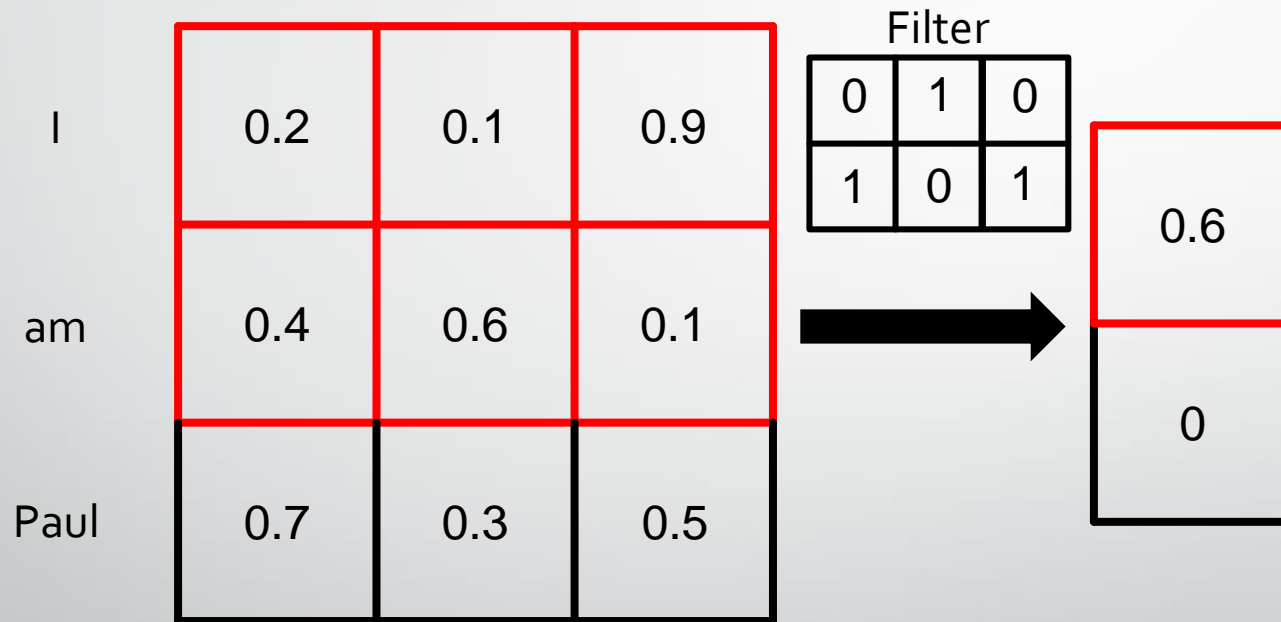
## Cons

- Needs a lot of data
- Requires a fixed input size
- Probably Approximately Correct
- Not humanly interpretable
- Takes a single vector as input
- Not good at detecting geometrical features

# Convolutional Neural Network



# Convolutional Neural Network



# Convolutional Neural Network

- Harkous et al. published a paper where they used a CNN to classify data practices in Privacy Policies
- They achieved an accuracy of 85% on the OPP-115 Corpus
- <https://arxiv.org/abs/1802.02561>



# Convolutional Neural Networks

## Pros

- Learns the function from data
- Can in theory represent any function (not so in practise)
- Very versatile
- **+ Exceptionally good at detecting geometrical features in data**

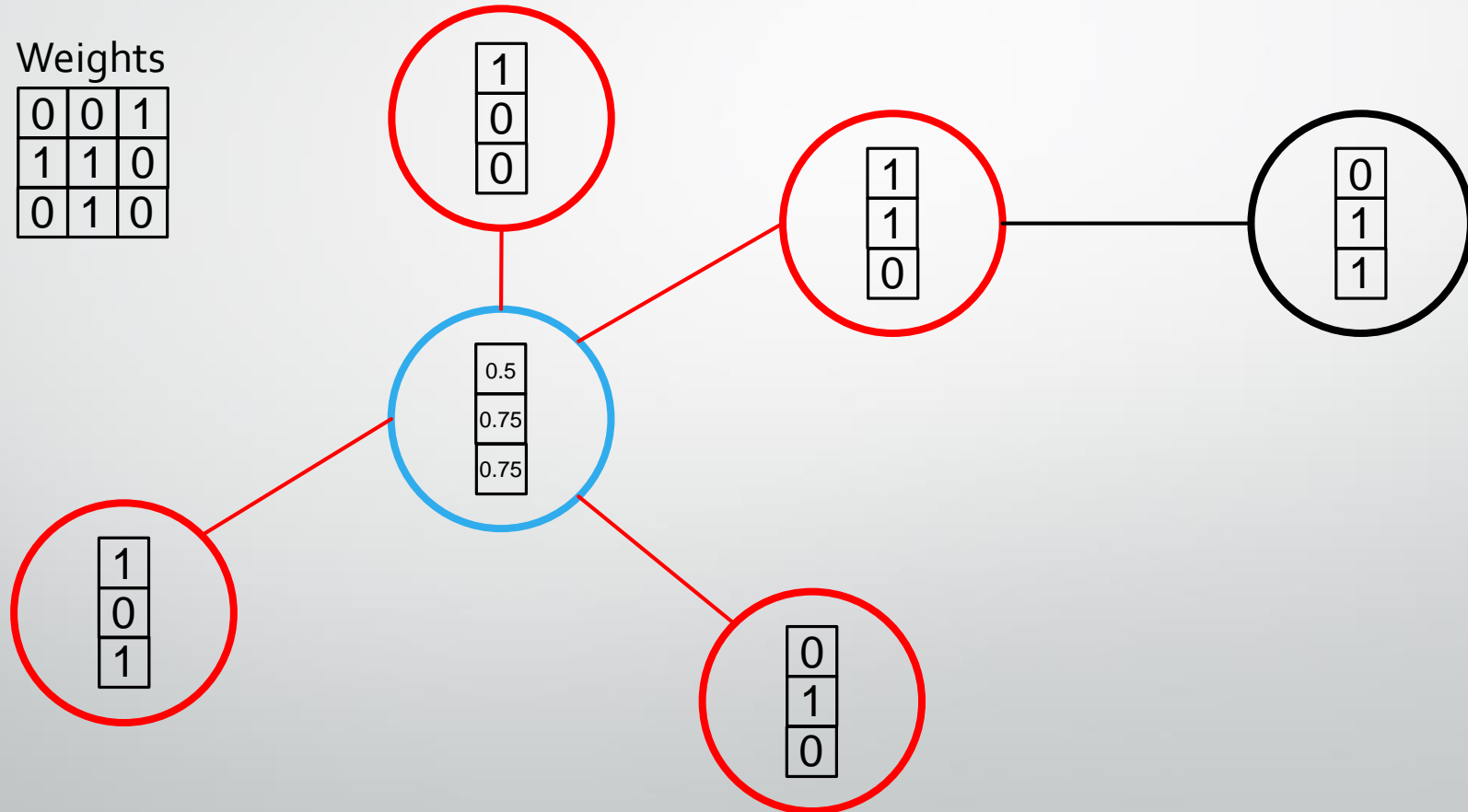
## Cons

- Needs a lot of data
- Requires a fixed input size
- Probably Approximately Correct
- Not humanly interpretable
- **+ Not good at relating features that are far apart**

# Graph Convolutional Neural Network

Weights

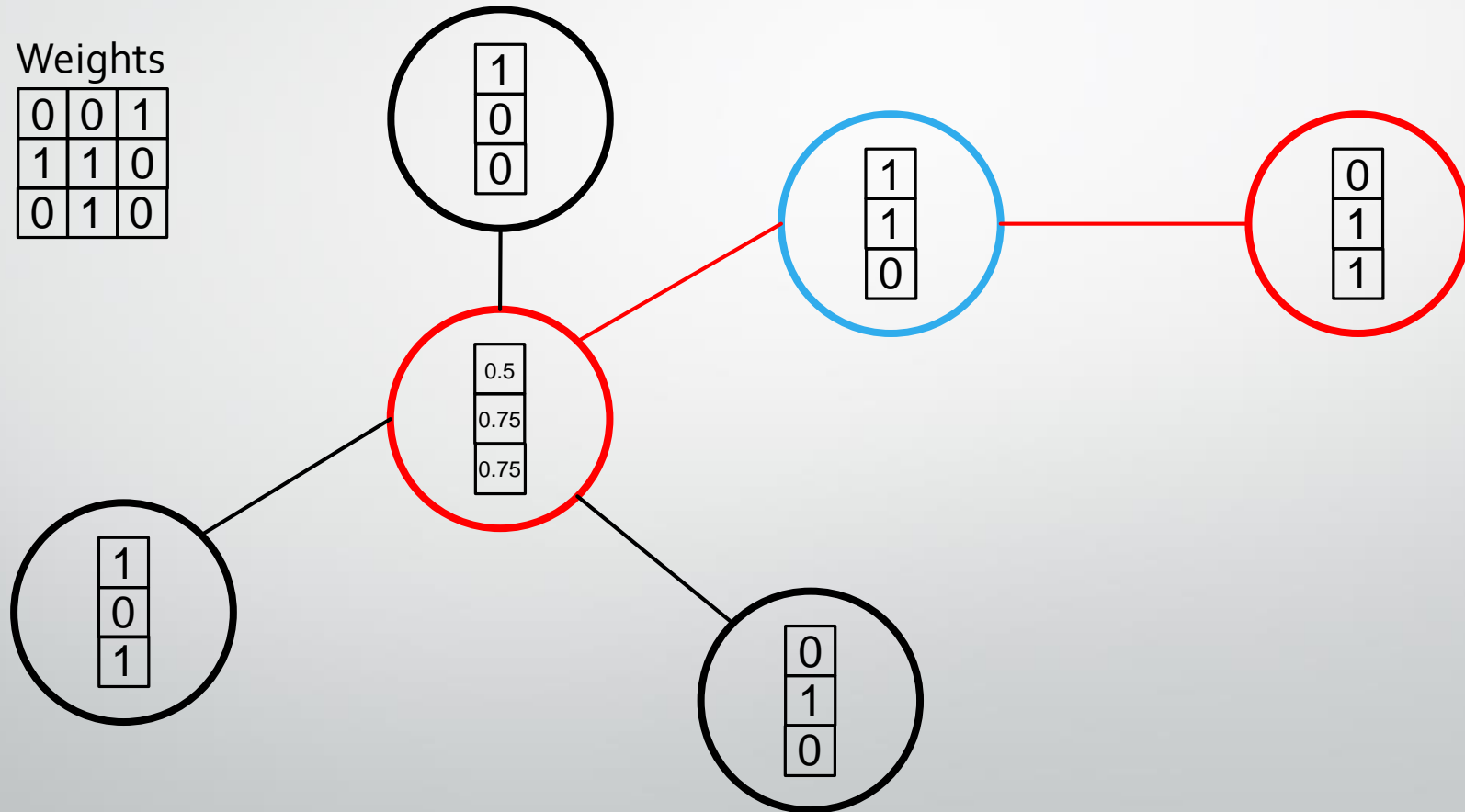
0	0	1
1	1	0
0	1	0



# Graph Convolutional Neural Network

Weights

0	0	1
1	1	0
0	1	0



# Graph Convolutional Neural Networks

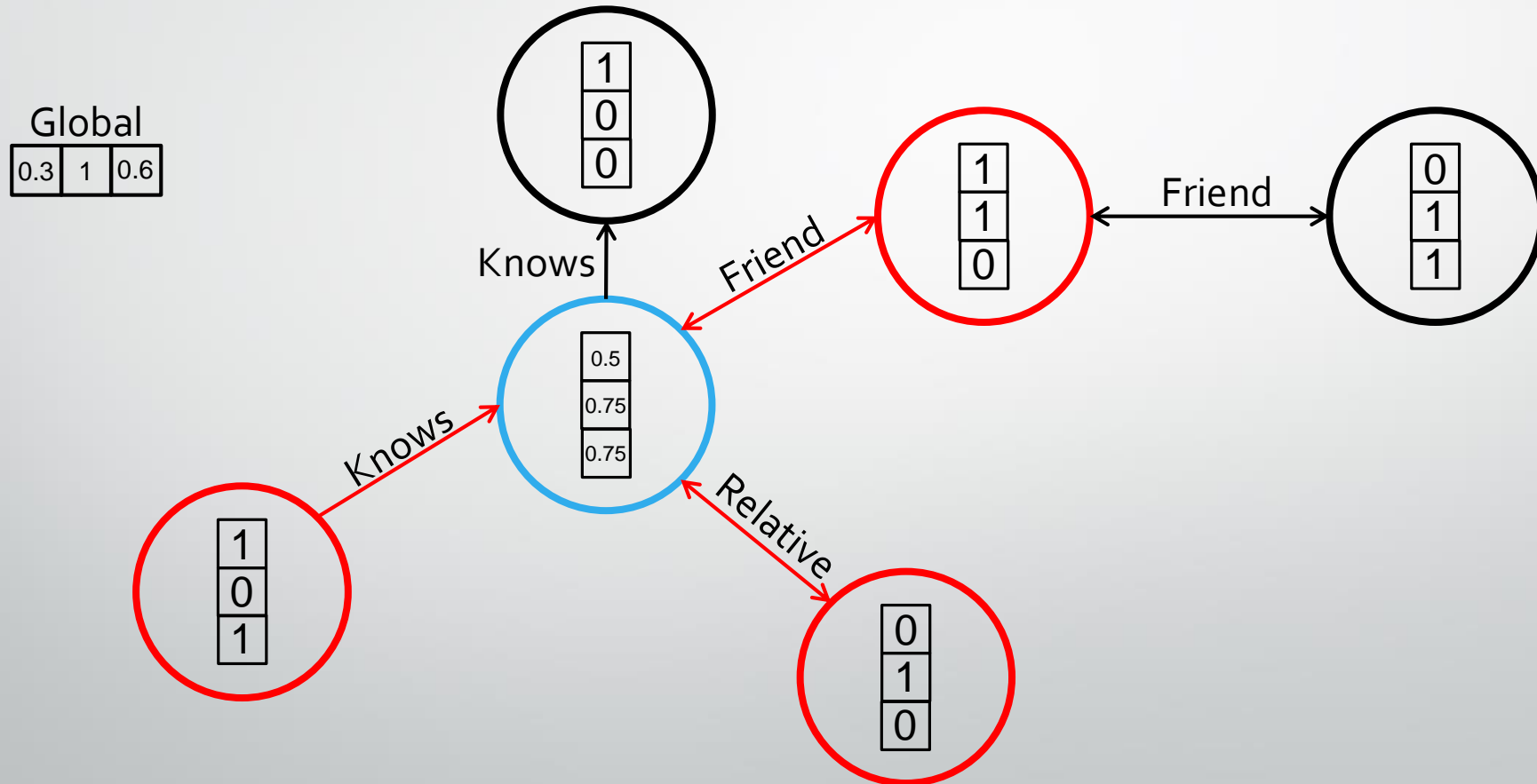
## Pros

- Learns the function from data
- Can in theory represent any function (not so in practise)
- Very versatile
- Good at detecting geometrical features in data
- **+ A lot of problems can be represented by graphs**
- **+ Is more robust to reduced amounts of data**
- **+ Works with variably sized input**
- **+ Edges can connect distant related features**

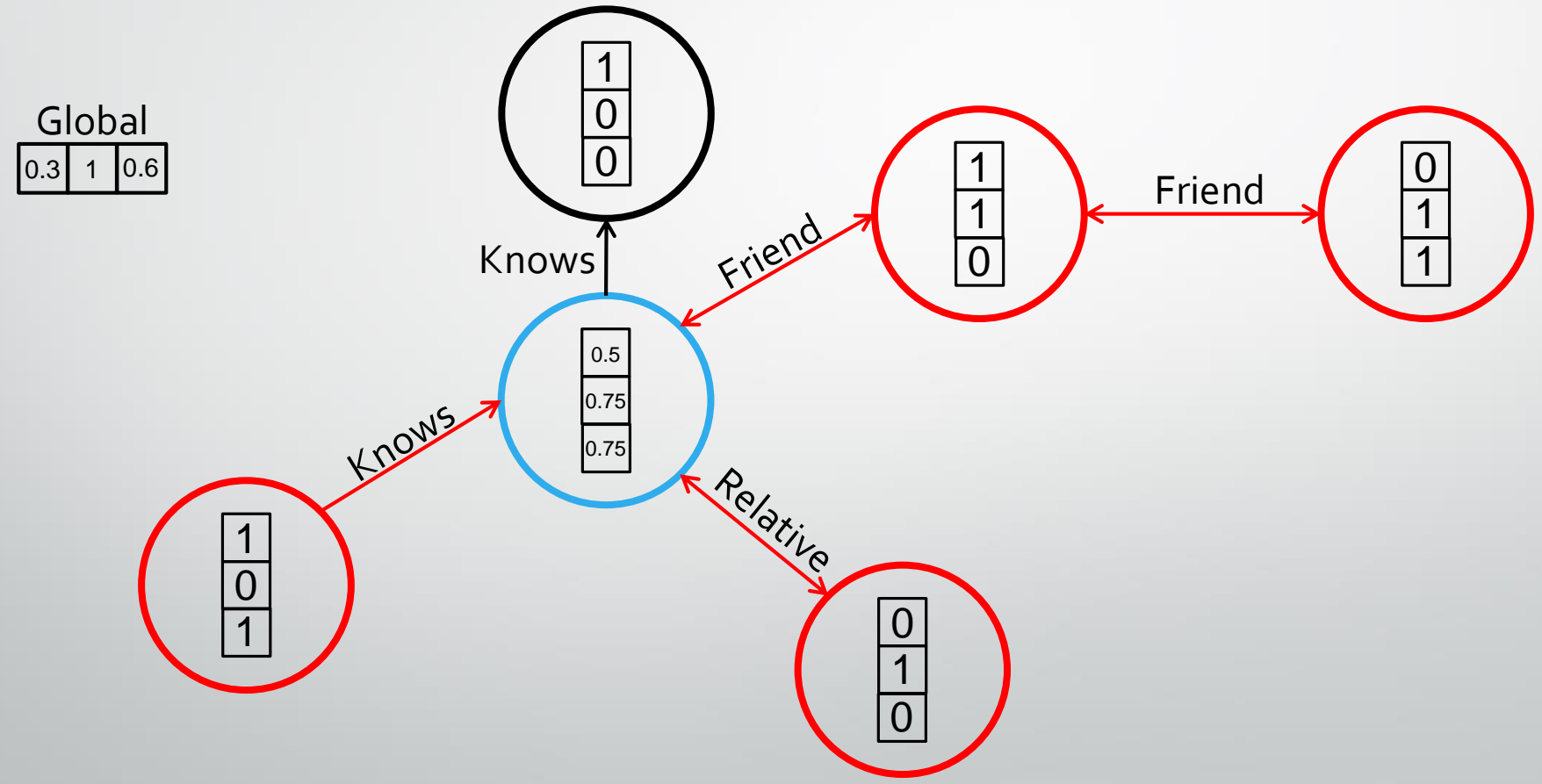
## Cons

- Still needs a fair bit of data
- Probably Approximately Correct
- Not humanly interpretable
- **+ Does not support edges of different types**

# Message Passing Graph Neural Network



# Message Passing Graph Neural Network



# Message-Passing Graph Neural Networks

## Pros

- Learns the function from data
- Can in theory represent any function (not so in practise)
- Very versatile
- Good at detecting geometrical features in data
- A lot of problems can be represented by graphs
- Is more robust to reduced amounts of data
- Works with variably sized input
- Edges can connect distant related features
- **+ Supports edges of different types**

## Cons

- Still needs a fair bit of data
- Probably Approximately Correct
- Not humanly interpretable



# Message Passing Graph Neural Network

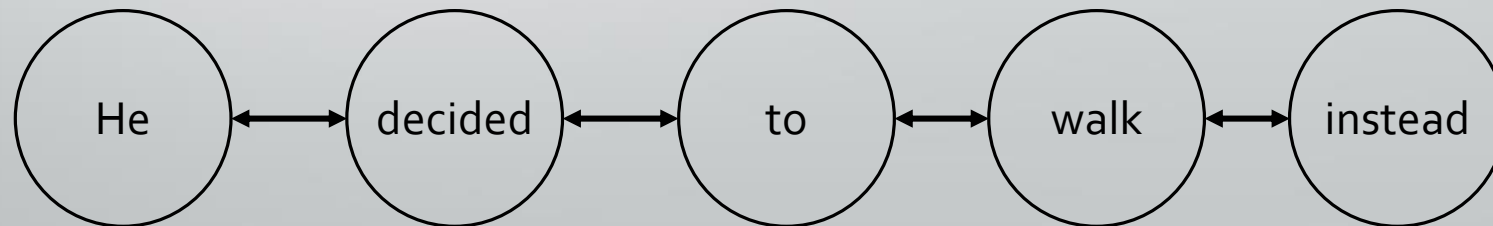
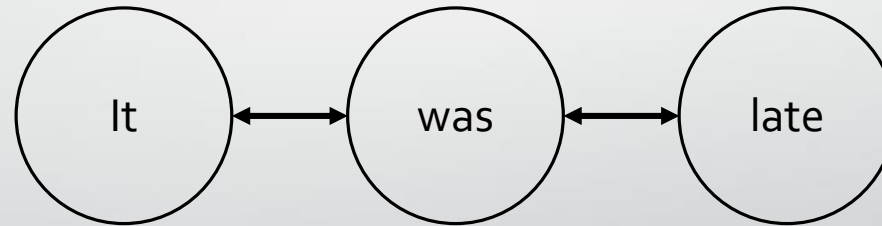
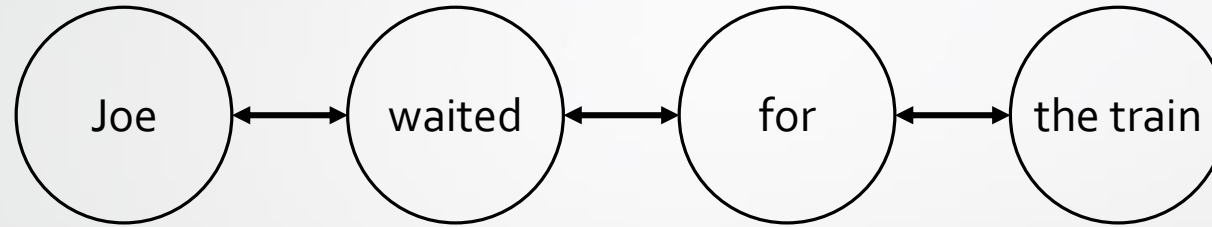
“Joe waited for the train.

It was late.

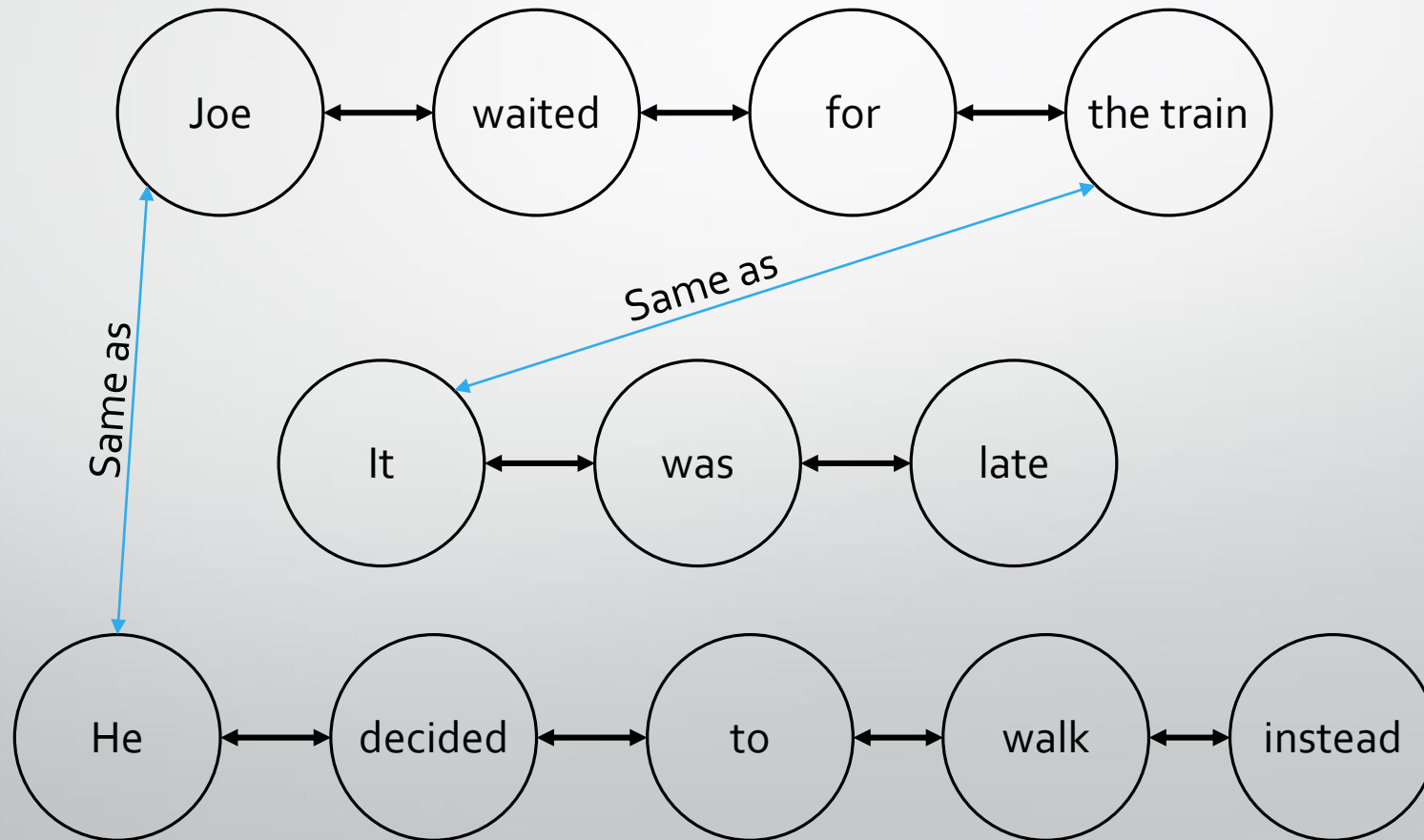
He decided to walk instead.”



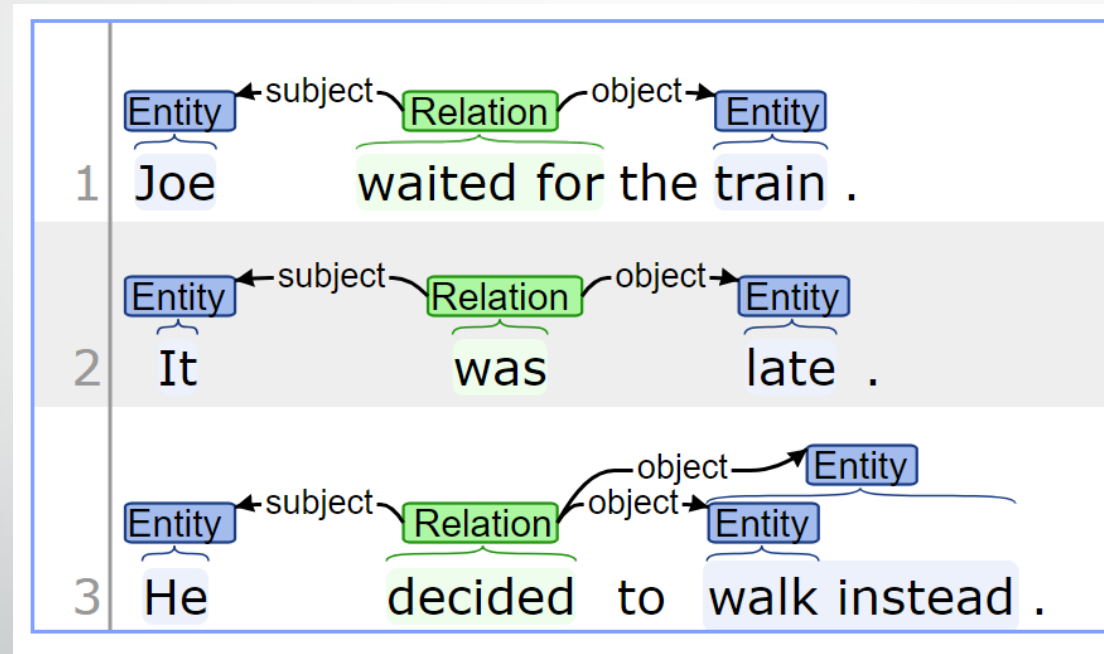
# Message-Passing Graph Neural Network



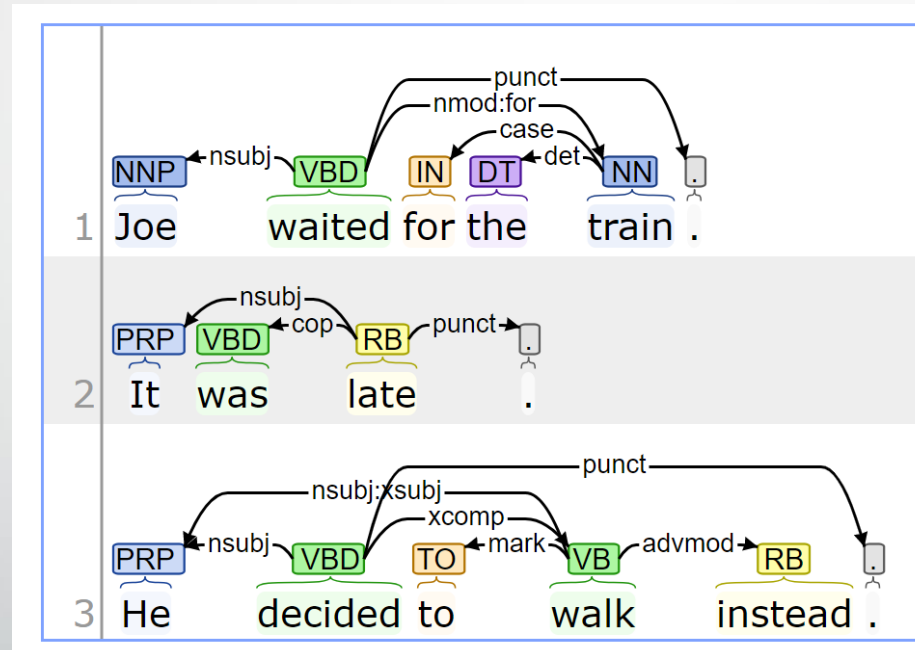
# Message-Passing Graph Neural Network



# Message-Passing Graph Neural Network



# Message-Passing Graph Neural Network



# Current work

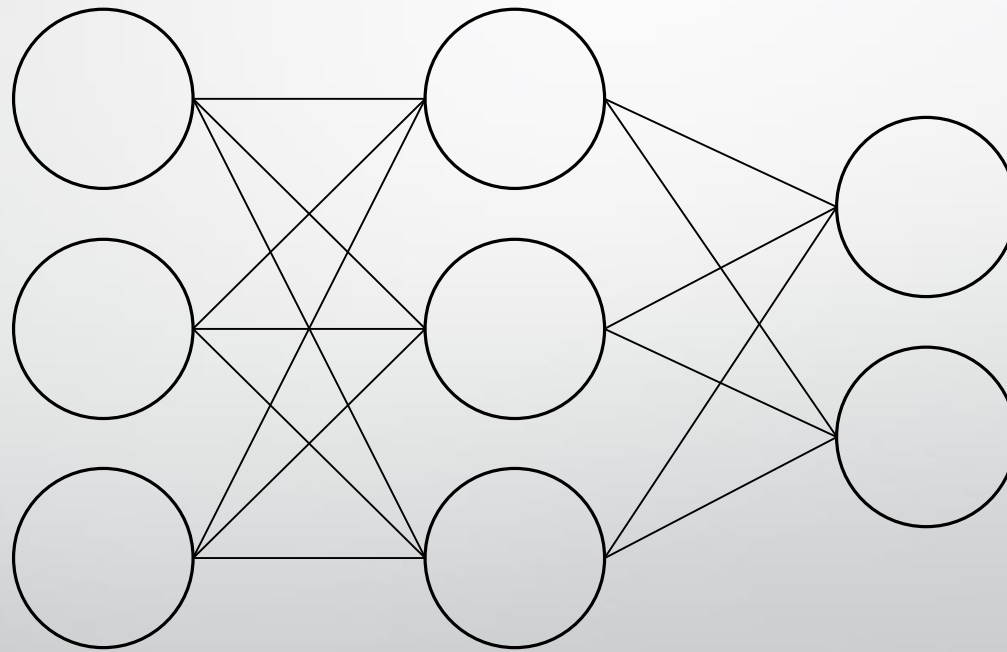
- Test Neural Network based tools on Privacy Agreement texts
  - GCN have not been used on the OPP-115 Corpus
  - MPGNN have not been used in general NLP
- Currently testing how well GCN and MPGNN perform on Privacy Agreement data sets
  - Compare to results from CNN
  - Privacy Agreement data sets are small and few
  - See whether additional Legal Knowledge can improve the results of MPGNN over other types of Neural Networks

Work with the University of Swansea

# Reference

- [\*The creation and analysis of a website privacy policy corpus\*](#). Shomir Wilson, Florian Schaub, Aswarth Abhilash Dara, Frederick Liu, Sushain Cherivirala, Pedro Giovanni Leon, Mads Schaarup Andersen, Sebastian Zimmeck, Kanthashree Mysore Sathyendra, N. Cameron Russell, Thomas B. Norton, Eduard Hovy, Joel Reidenberg, and Norman Sadeh. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, August 2016*.

# Neural Networks



# Neurons

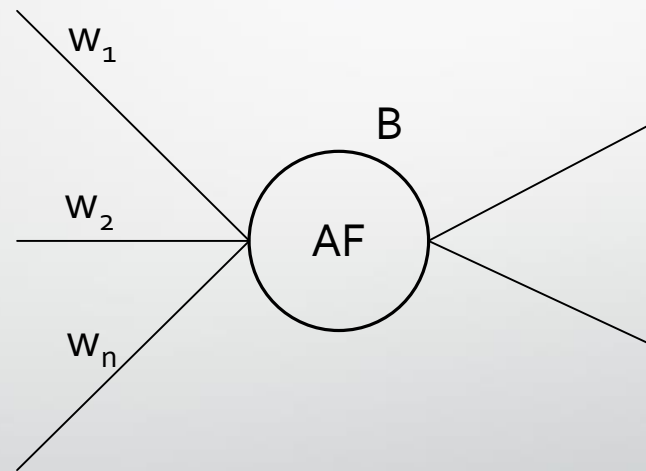
W – Weight

B – Bias

AF – Activation Function

*Output =*

$$AF(B + \sum_n input_n \times wn)$$

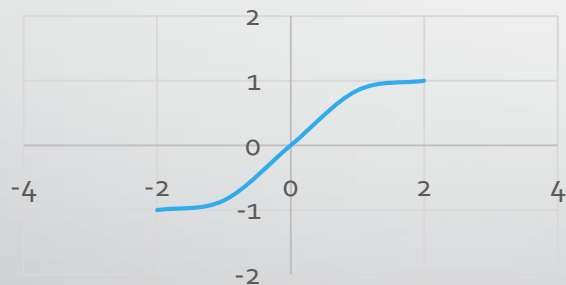




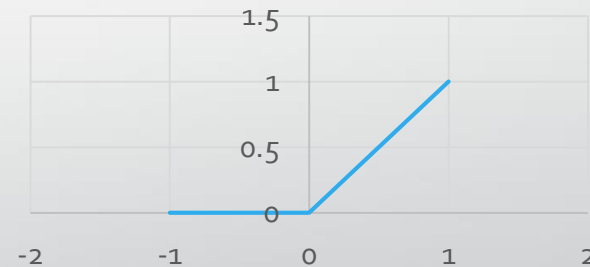
# Activation Functions

- Sigmoid
- Rectified Linear Unit (ReLU)

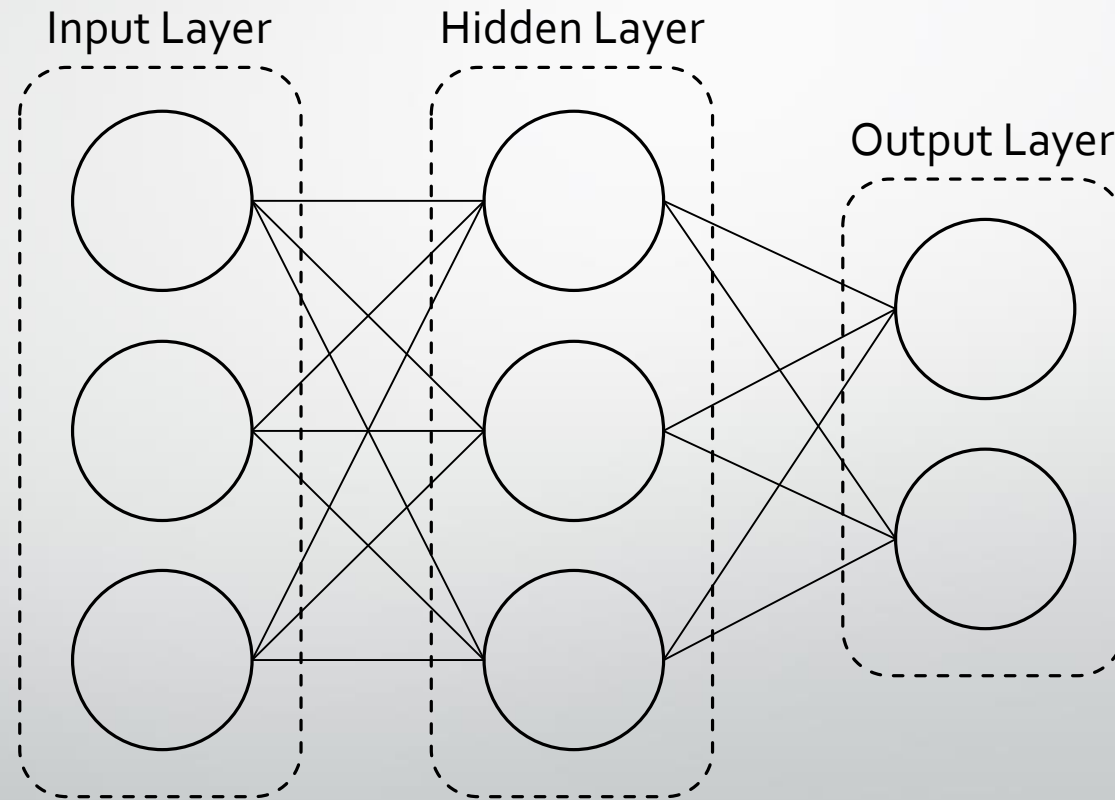
Sigmoid



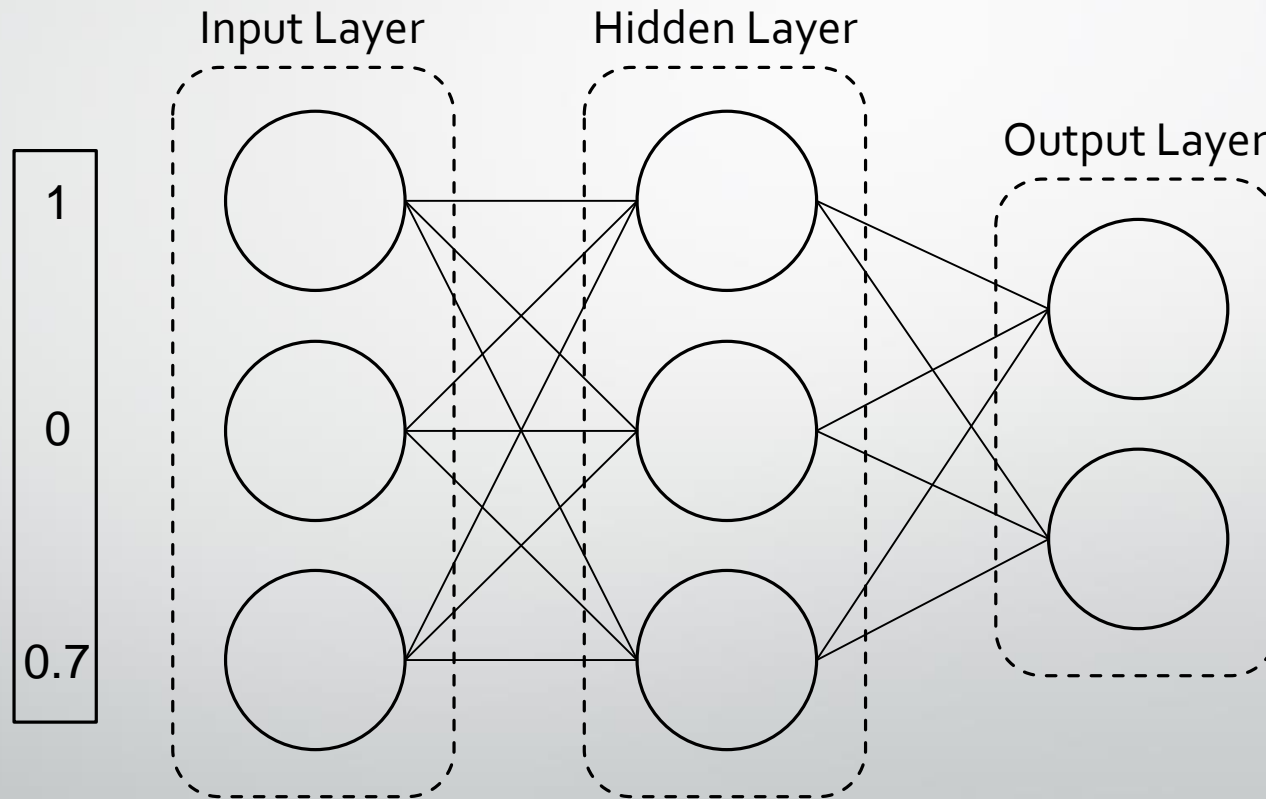
ReLU



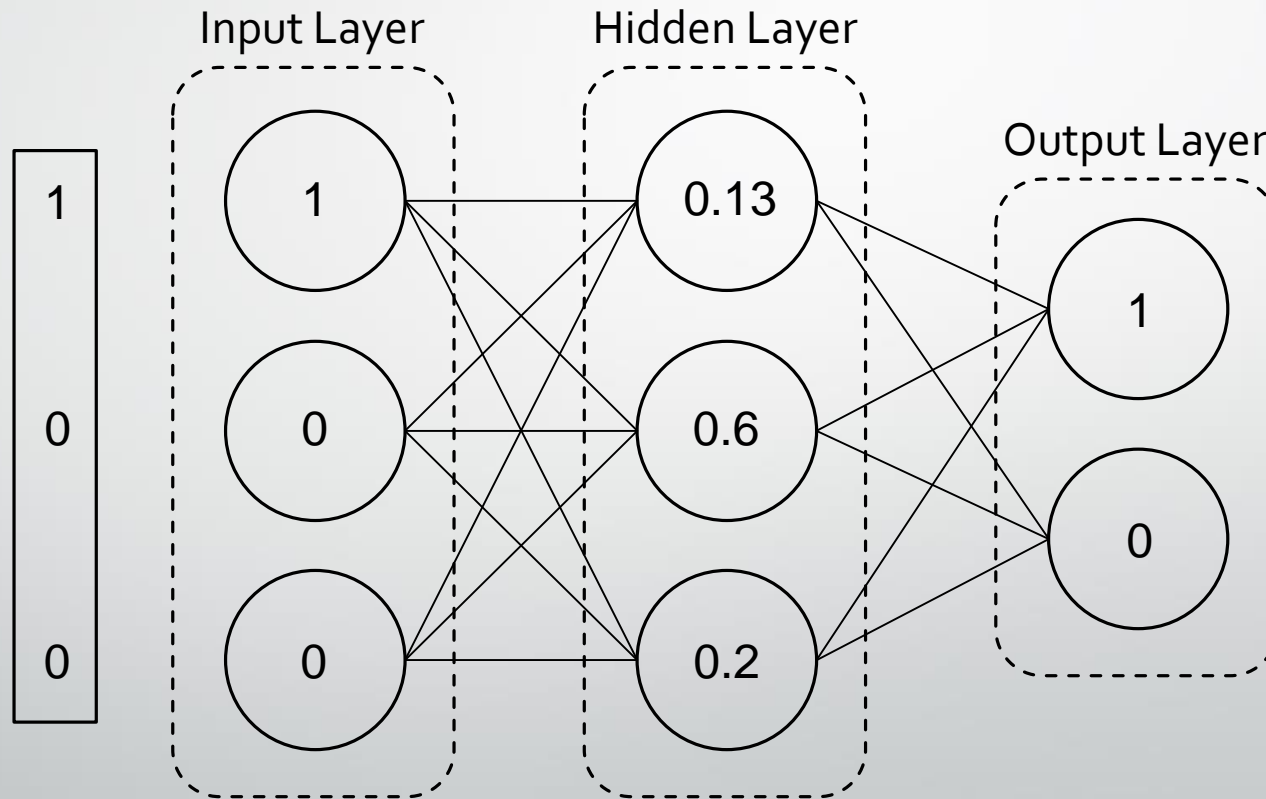
# Neural Networks



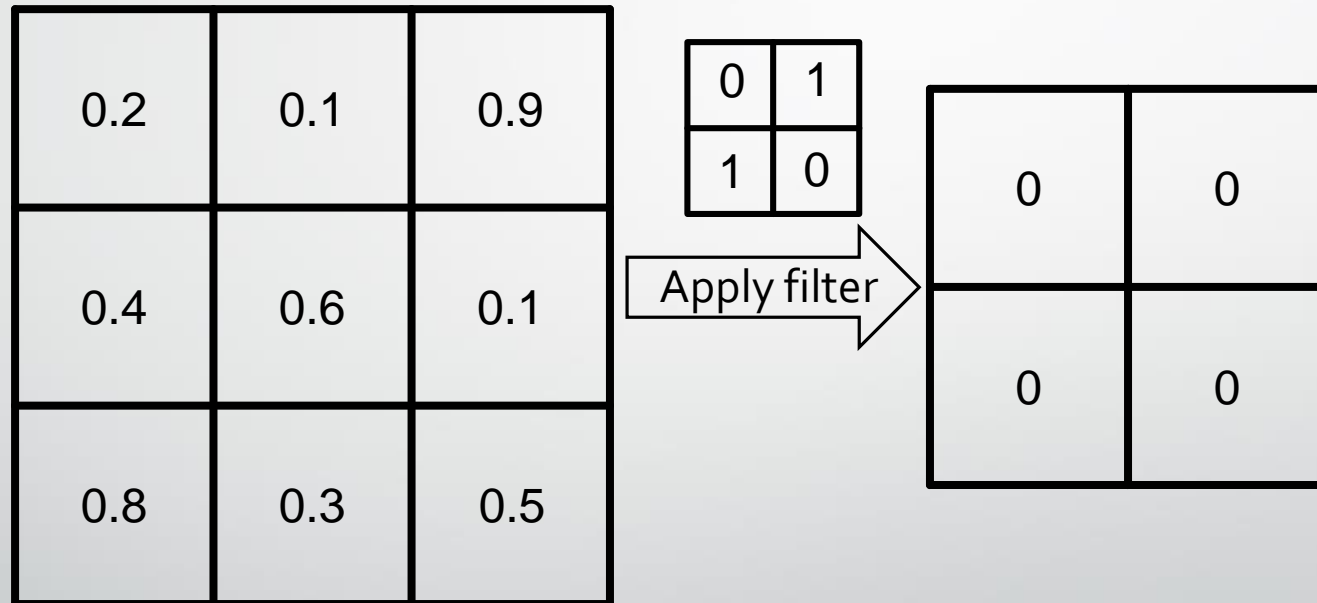
# Neural Networks



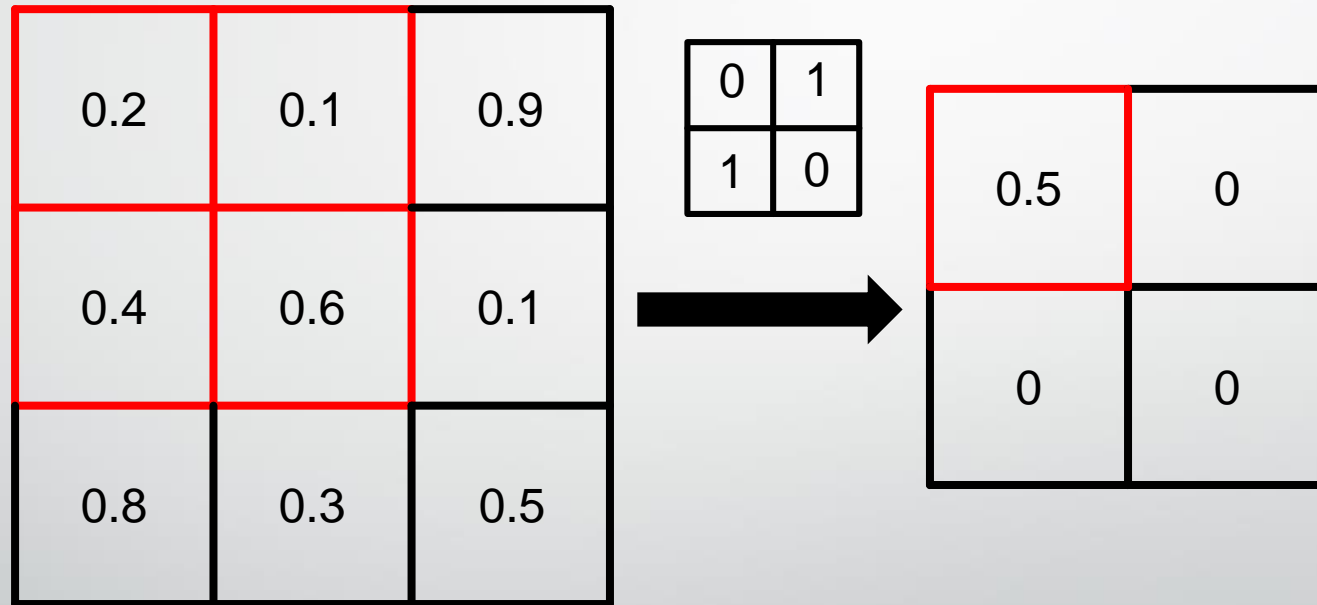
# Neural Networks



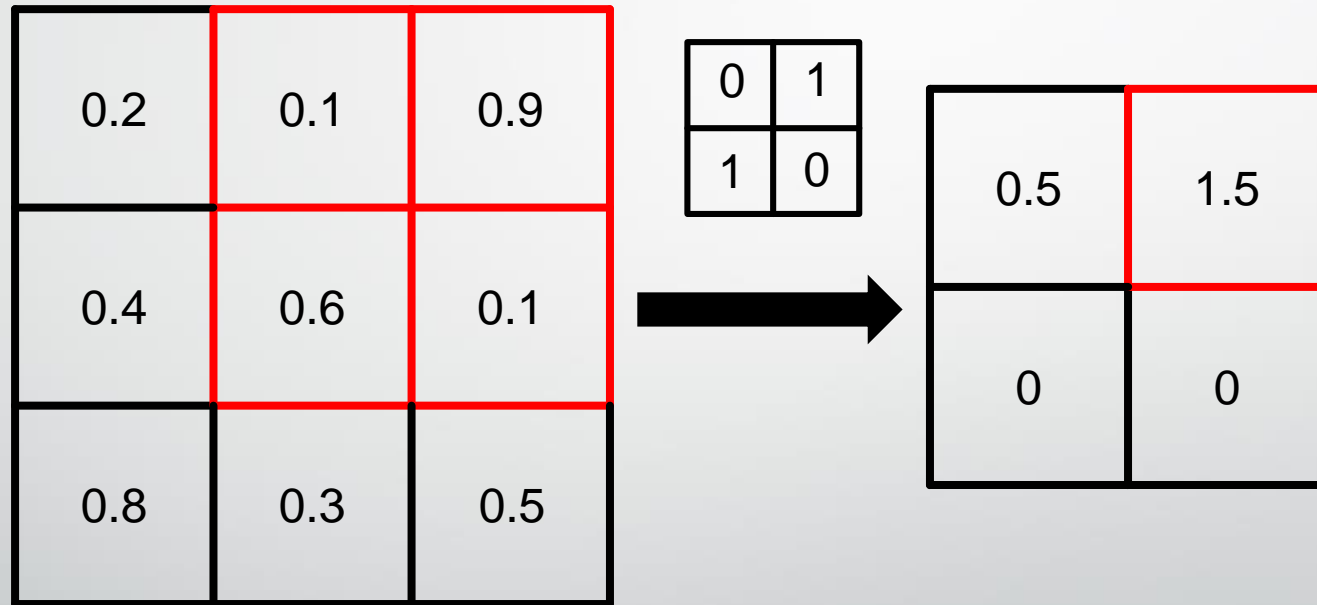
# Convolutional Neural Network



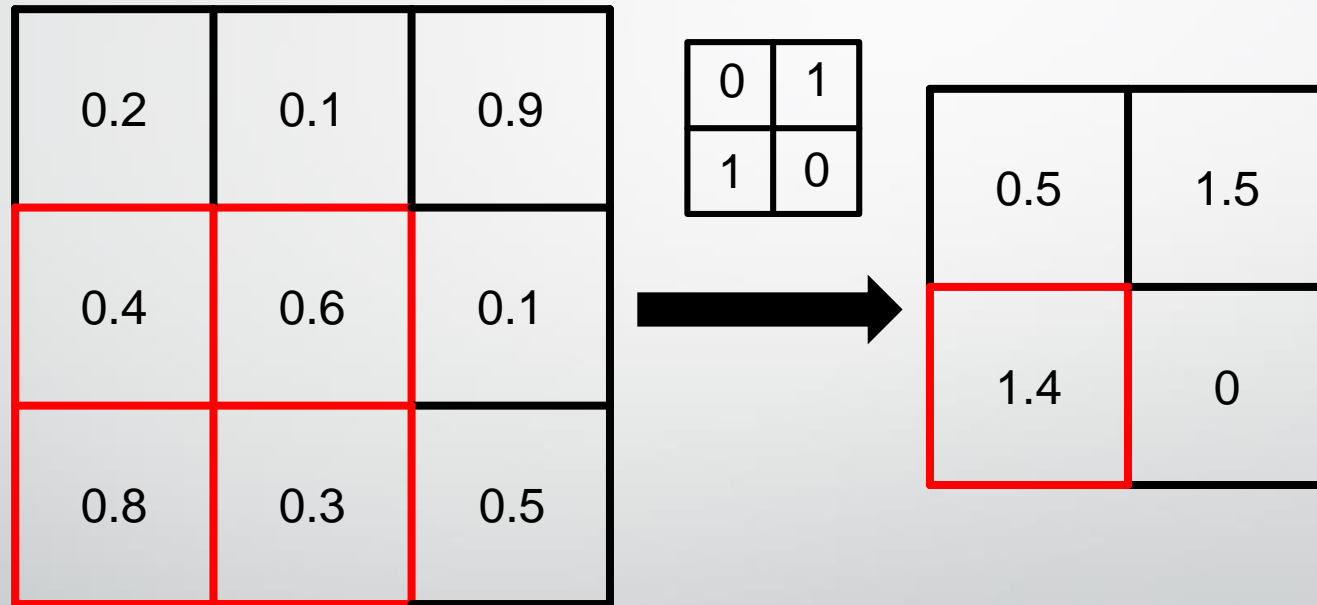
# Convolutional Neural Network



# Convolutional Neural Network

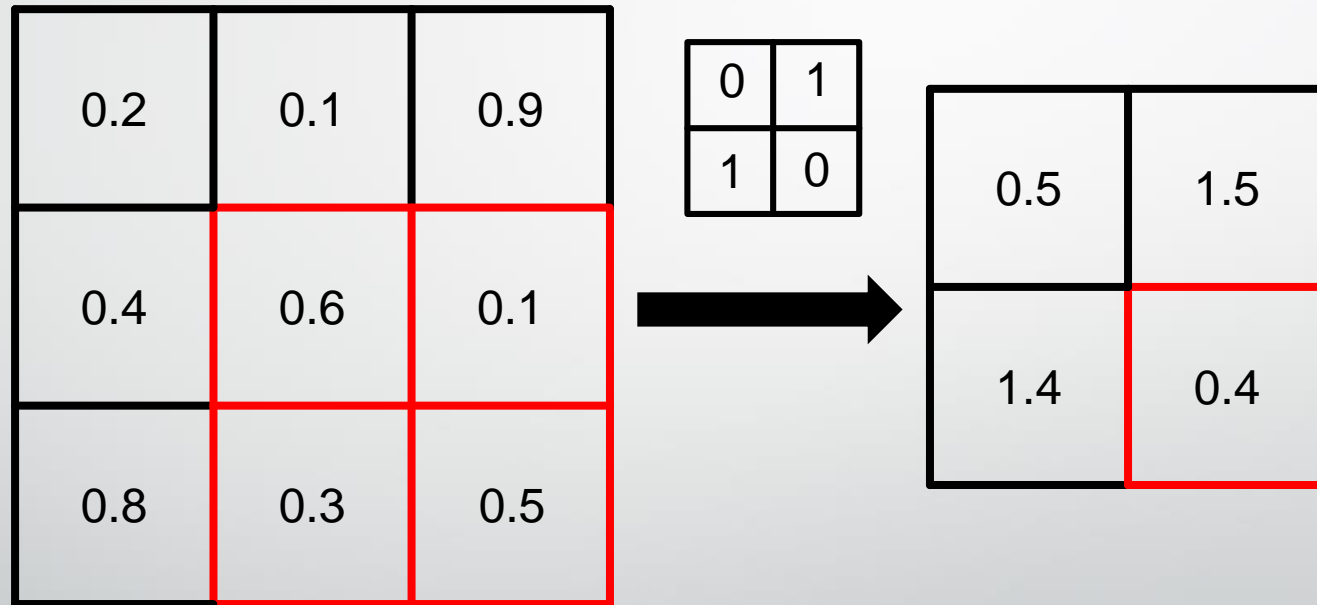


# Convolutional Neural Network





# Convolutional Neural Network

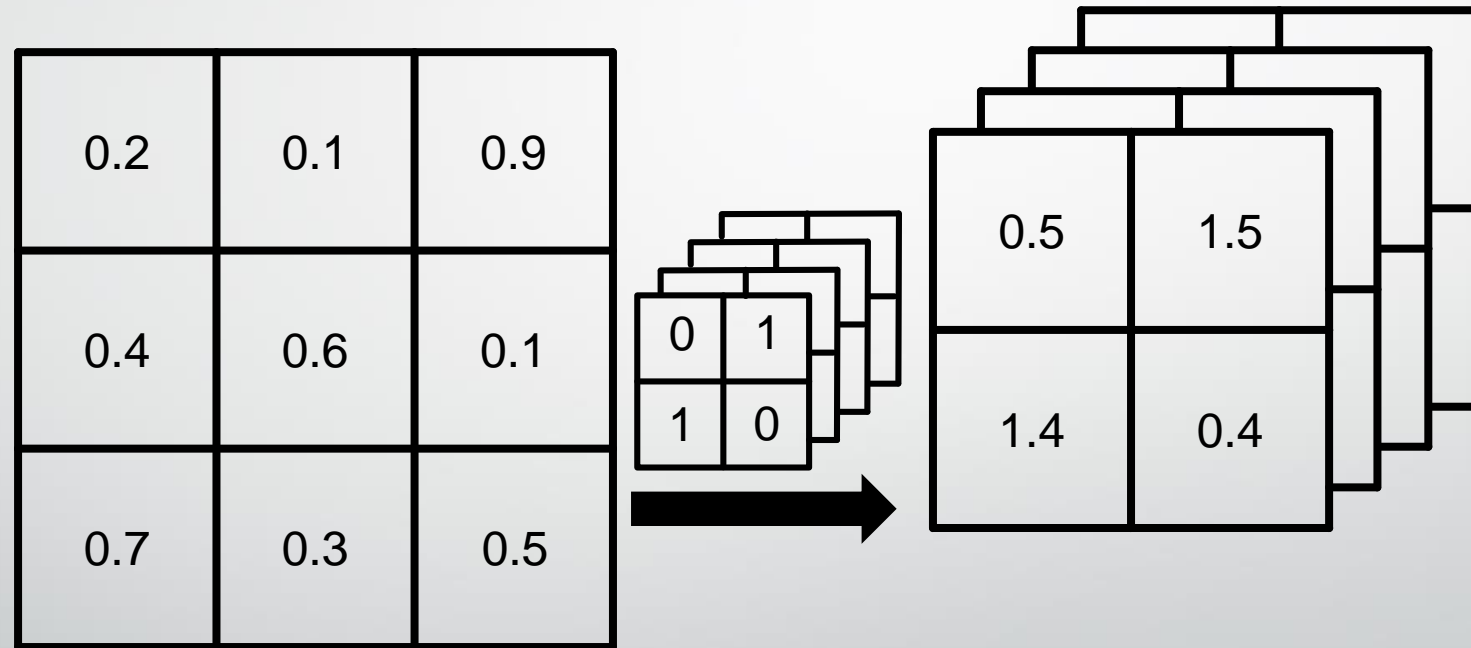


# Convolutional Neural Network

0.2	0.1	0.9
0.4	0.6	0.1
0.7	0.3	0.5

0.5	1.5
1.4	0.4

# Convolutional Neural Network



# Convolutional Neural Network



# Convolutional Neural Network



# Convolutional Neural Network

