SRH Hochschule Heidelberg

Analytics 4

Dealing with Textual data

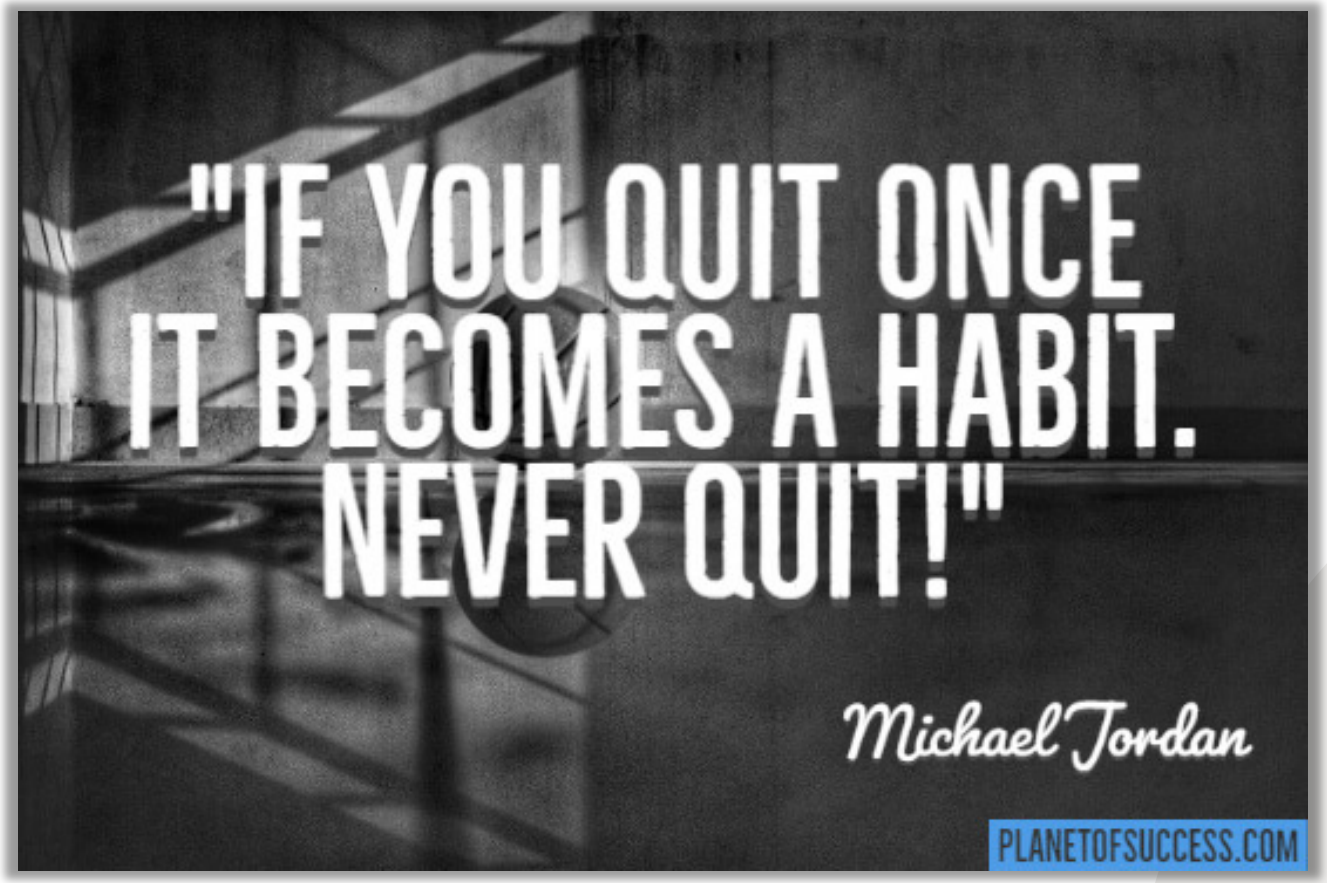
Academic Researcher: Ashish Chouhan
External Dozent: Ajinkya Patil

Date of Lecture: 09.06.2021

Natural Language Processing (NLP)

01

Natural Language Processing Example



https://colab.research.google.com/drive/1Fc0sjD1k2A7-0bjtRgwzd8pvgQTAmgMe#scrollTo=hPgOY0hcUKOm

What is Natural Language Processing?

- ***Natural Language Processing (NLP) is a method or a technique to perform Text Mining***
- NLP is a special kind of linguistics analysis that helps a machine to **understand text**
- NLP application: https://github.com/sebastianruder/NLP-progress
- For NLP, a consistent knowledge base is required
  - Theasaurus
  - Lexicon of words
  - Grammatical rules

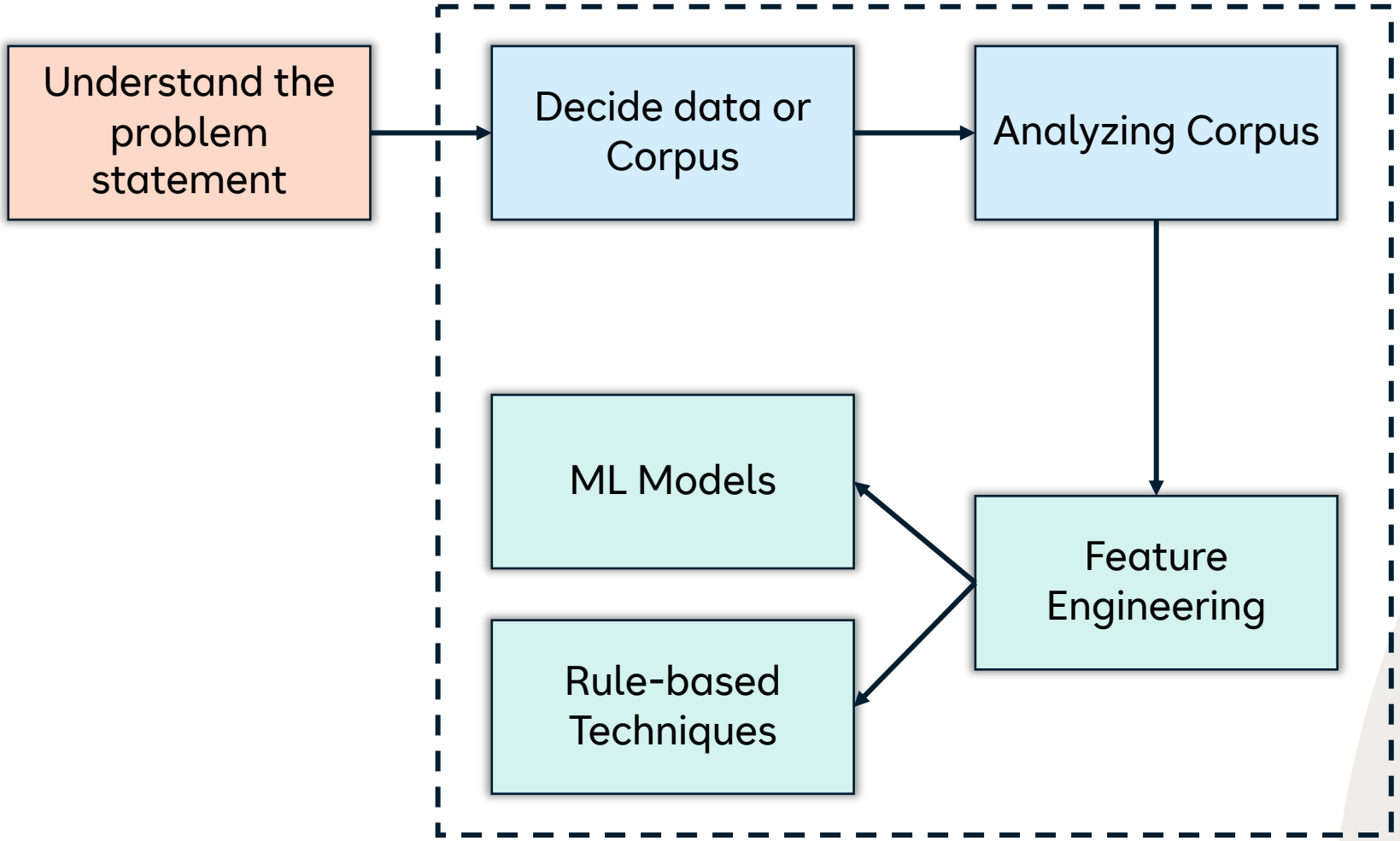
Text Mining V/s Natural Language Processing

Points	Text Mining	Natural Language Processing
Goal	Conversion of textual content into data for further analysis	Computer systems can understand human language or text
Data Sources	Document Collection	Any form of Natural Human Communication like text, speech, signboard and so on
Perform task	Does **NOT** consider Semantic Analysis	Consider Syntactic and Semantic Analysis
Human Intervention	Does not require human intervention	Does require human intervention
Output	Patterns of words, Frequency of words, Correlation within words	Conveyed sentiment, Semantic meaning of text, Grammatical sturcture
Tools	Statistical Models, ML models	Advanced ML models, Deep Neural Networks

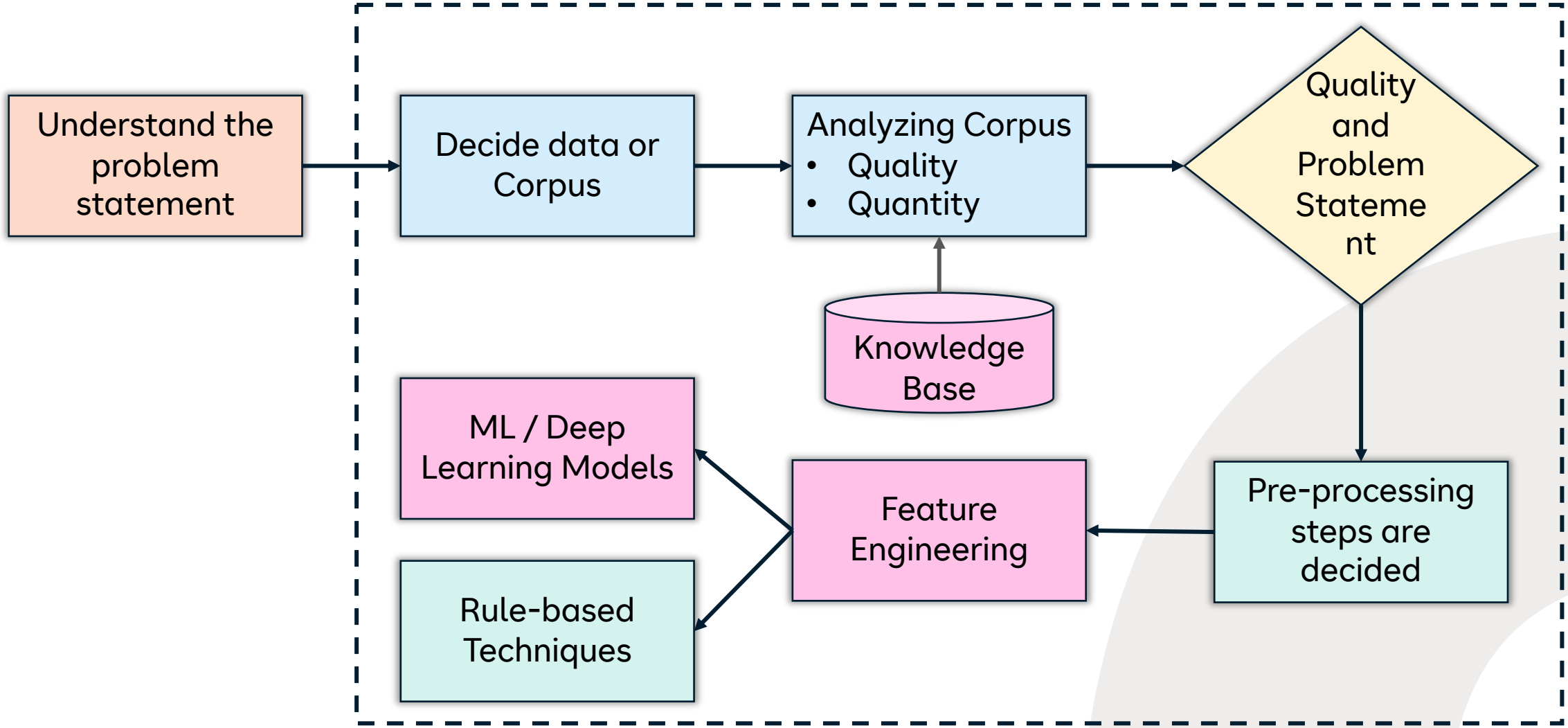
Text Mining V/s Natural Language Processing

Points	Text Mining	Natural Language Processing
Applications	• Contextual Advertising • Spam filtering • Fraud detection • **Information Extraction** • **Information Retrieval**	• Speech recognition system • Question answering system • Language translation • Text summarization • Sentiment analysis • Chatbots • Text Classification • Topic Segmentation
Summarize	About deriving the information from text	Teaching a computer to recognize, understand and process human speech

Life Cycle of Text Mining



Life Cycle of Natural Language Processing



Textual data (Text)

Why dealing with Text is Tough? [Hearst , 1997]

- **Abstract concepts** are difficult to represent
- "Countless" combinations of **subtle, abstract relationships** among concepts
- Many ways to **represent similar concepts**
  - Example: Spaceship, flying saucer, UFO
- High dimensionality
- Tens or hundred of thousands of features

Why dealing with Text is Easy? [Hearst , 1997]

- Highly **redundant** data
- Simple Algorithms can also get "good" results for simple tasks:
  - Pull out " important" **phrases**
  - Find "**meaningfully**" related words
  - Create some sort of **summary** from documents

How do we represent Text?

Levels of text representations

Lexical Level:

- Character (character n-grams and sequences)
- Words (Stop-words, stemming, lemmatization)
- Phrases
- Taxonomies

Levels of text representations

Syntactic Level:

- Vector-space model
- Language models
- Full-parsing
- Cross-modality

Levels of text representations

Semantic Level:

- Collaborative tagging
- Templates
- Ontologies

Lexical Level

Lexical Level

- **Character (character n-grams and sequences)**
- Words (Stop-words, stemming, lemmatization)
- Phrases
- Taxonomies

Character Level

- Character level representation of a text consists of **sequence of characters**
  - A document is represented by a frequency distribution of sequences
  - Each character sequence of length 1,2,3,… represent a feature with its frequency

- Strengths:
  - Very Robust since it avoids language **morphology** (Example: Language identification)
  - Captures simple **patterns on character level** (Example: Spam Detection, Copy Detection)

- Weakness:
  - **Deeper Semantic Tasks**, the representation is too weak (Example: Understanding the text)
  - As these are not words but just a sequence of characters

Lexical Level

- Character (character n-grams and sequences)
- **Words (Stop-words, stemming, lemmatization)**
- Phrases
- Taxonomies

Word Properties

Word is the most common representation of text used for many techniques

- **Homonomy:**
  - Same form, but different meaning (Example - Bank: Riverbank, Financial Institution)
- **Polysemy:**
  - Same form, related meaning (Example – Bank: Blood bank, Financial Institution)
- **Synonymy:**
  - Different form, same meaning (Example - Singer, Vocalist)
- **Hyponymy:**
  - One word denotes a subclass of an another (Example: Breakfast, Meal)

Word Properties

- Word frequencies in texts have **power distribution:**
  - Small number of very frequent words (Example: the, an, a, of, to..)
  - Big number of low frequency words

Word Level

Tokenization:

- **Sentence Tokenization:**
  - Split the document or paragraph into sentence
- **Word Tokenization:**
  - Split the text into words

Lowercasing:

- Convert the word into lowercase
- **Important: Not always converting into lowercase is recommended**

Word Level

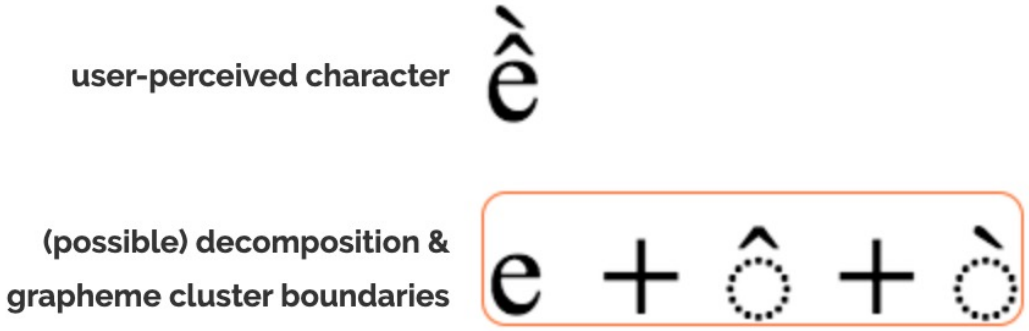
Stop-words Removal:

- Stop-words are the words that from non-linguistic do not carry much information
  - They have mainly functional role
  - Remove them to help the methods to perform better
- Stop words are **language dependent**
  - English: A, About, Above, Across, After, Again, ….
  - Deutsch: über, unter, nach, von, ….
  - Dutch: de, en, van, ik, te, …

Word Level

Word and Character level Normalization:

- Issues that we usually avoid:
  - Plenty of character encodings in use, often nontrivial to identify a word and write it in unique form.
  - Example: In Unicode the same word could be written in many ways



https://www.w3.org/International/articles/definitions-characters/

Word Level

Stemming:

- Different forms of the same word are usually problematic for text data analysis, because they have different spelling and similar meaning (Example: Learns, Learned, Learning….)
- Stemming is a process of transforming a word into its **stem** (root form)
- Porter Stemmer - https://tartarus.org/martin/PorterStemmer/
- Snowball - http://snowball.tartarus.org/ (https://github.com/snowballstem/snowball)

Example of rules used in English Porter Stemmer:

- ATIONAL -> ATE (Example: relational -> relate)
- TIONAL -> TION (Example: conditional -> condition)
- IZER -> IZE (Example: digitizer -> digitize)
- ENTLY -> ENT (Example: differently -> different)

Word Level

Lemmatization:

- Different forms of the same word are usually problematic for text data analysis, because they have different spelling and similar meaning (Example: Learns, Learned, Learning….)
- Lemmatization is a process of transforming a word into its **lemma** (root form)
- Word Net Lemmatizer - https://wordnet.princeton.edu/

Example:

Words:	run	running	ran	runs
Stemming	run	run	**ran**	run
Lemmatization	run	run	**run**	run

Word Level

Part-of-Speech (PoS) Tags:

- Part-of-Speech tags is for annotating the word's function
- Use of Part-of-Speech:
  - Information Extraction (Example: Interested in only named entities that are “noun phrases”
  - Reduction of the Vocabulary (Features) (Example: Most information is carried by nouns)
- Part-of-Speech taggers are usually learned by Hidden Markov models algorithm on manually tagged data.
- Part-of-Speech Table: https://www.englishclub.com/grammar/parts-of-speech.htm

Lexical Level

- Character (character n-grams and sequences)
- Words (Stop-words, stemming, lemmatization)
- **Phrases**
- Taxonomies

Phrases Level

- Instead of just single words we can deal with phrases
- Main effect of using phrases is to more precisely identify sense and get rid of synonyms

N-grams:

- N-Gram Corpus: https://www.ngrams.info/compare.asp
- N-Gram Viewer: https://infoguides.gmu.edu/textanalysistools/ngram
- Blog: https://ai.googleblog.com/2006/08/all-our-n-gram-are-belong-to-you.html#links

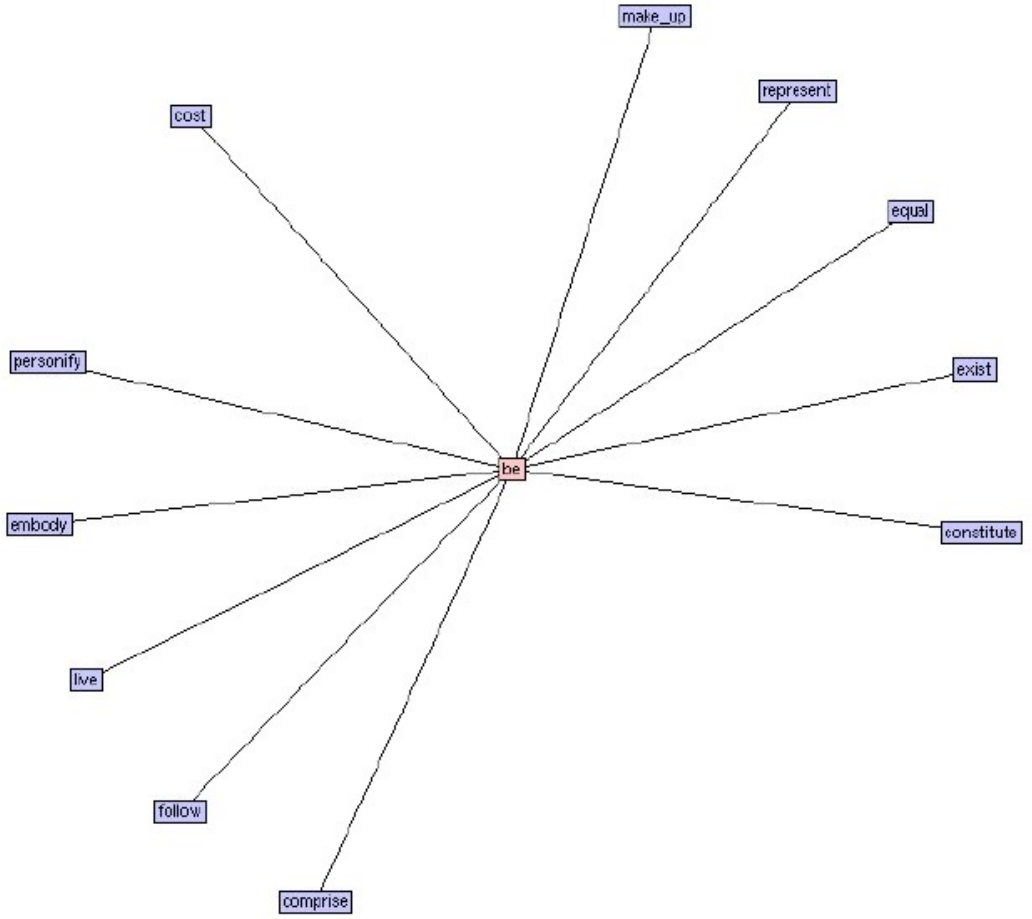
Lexical Level

- Character (character n-grams and sequences)
- Words (Stop-words, stemming, lemmatization)
- Phrases
- **Taxonomies**

Taxonomies Level

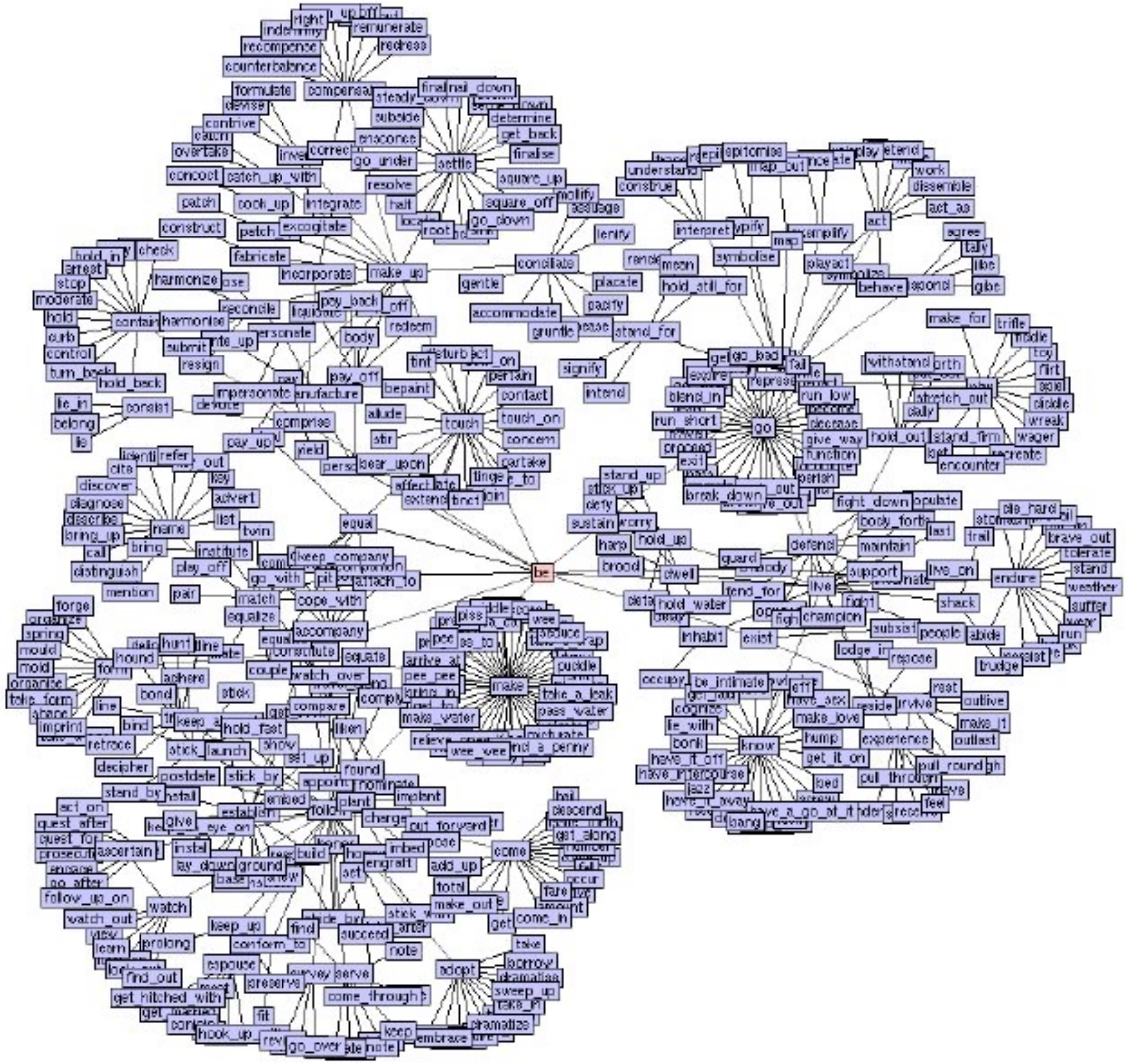
- Main function of Taxonomies is to connect different surface word forms with the same **relationships** (Synonyms, hypernyms are connected)
- Commonly used general Thesaurus is **WordNet** present in different languages
- WordNet [Miller et al., 1990] : https://wordnet.princeton.edu/
  - Database of Lexical relations
  - Consist of different databases (Noun, Adjectives, Verbs, Adverbs)
  - Each database consist of sense entries : each sense consist of a set of synonyms (Example: person, individual, someone; musician, instrumentalist, player)
  - WordNet Online: http://wordnetweb.princeton.edu/perl/webwn
- EuroWordNet: https://archive.illc.uva.nl//EuroWordNet/
  - 8-10 languages are covered

WordNet



[Kamps et al., 2002]

WordNet



[Kamps et al., 2002]

WordNet

- Each WordNet entry is connected with other entries in the graph through relations
- Relations in the database of nouns:

Relations	Definition	Example
Hypernym	From lower to higher concepts	Brekfast -> meal
Hyponym	From concepts to subordinates	Meal -> lunch
Has-Member	From groups to their members	Faculty -> Professor
Member-Of	From members to their groups	Copilot -> Crew
Has-Part	From wholes to parts	Table -> leg
Part-Of	From parts to wholes	Course -> meal
Antonym	Opposites	Leader -> Follower

Syntactic Level

Syntactic Level

- **Vector-space model**
- Language models
- Full-parsing
- Cross-modality

Vector-space model Level

Most Common way to deal with documents/text/words is first to **transform into numeric vectors** (Sparse vectors and then deal with them with linear algebra)

- As soon as we do this, we forget the **linguistic structure** within the text. Also called as "Structural Curse" because this way of forgetting about the structure does not harm efficiency of solving many relevant problems.
- Representation is referred to as "Bag-of-Words" or " Vector-Space Model"

Example

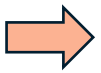
Sentence_1 => He is a good boy.

Sentence_2 => She is a good girl.

Sentence_3 => Boy and girl are good.

Text Pre-processing:

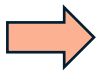
- Lowercasing
- Stemming
- Lemmatization
- Stop words removal



Sentence_1 => good, boy

Sentence_2 => good, girl

Sentence_3 => boy, girl, good



Words	Frequency
good	3
boy	2
girl	2

Bag-of-Words

Derive Vectors using Bag-of-Words (BoW):

1. Binary BoW => Where each word in the sentence is just represented as 0s or 1s
2. BoW => Where if one word is repeated more than 1 in a sentence then it is represented by the count value

Sentence	good (feature 1)	boy (feature 2)	girl (feature 3)
Sentence_1	1	1	0
Sentence_2	1	0	1
Sentence_3	1	1	1

Bag-of-Words

Disadvantages of BoW:

- Values are 1 or 0 (**Semantic is the same**, i.e., which word is having more importance is not present)
- Equal Weightage is given to all the words present in the sentence as they are represented by one value

Sentence	good (feature 1)	boy (feature 2)	girl (feature 3)
Sentence_1	1	1	0
Sentence_2	1	0	1
Sentence_3	1	1	1

Term Frequency- Inverse Document Frequency (Tf-IDF)

Derive Vectors using Tf-IDF:

$$Term\ Frequency = \left(\frac{Number\ of\ repetition\ of\ words\ in\ a\ Sentence}{Number\ of\ words\ in\ a\ Sentence}\right)$$

Words	Sentence_1	Sentence_2	Sentence_3
good	1/2	1/2	1/3
boy	1/2	0	1/3
girl	0	1/2	1/3

Term Frequency- Inverse Document Frequency (Tf-IDF)

Derive Vectors using Tf-IDF:

$$Inverse\ Document\ Frequency = \log\left(\frac{Number\ of\ Sentences}{Number\ of\ Sentence\ containing\ that\ word}\right)$$

Words	IDF	IDF Values
good	Log(3/3)	0
boy	Log(3/2)	0.176
girl	Log(3/2)	0.176

Term Frequency- Inverse Document Frequency (Tf-IDF)

Derive Vectors using Tf-IDF:

$$Tf - IDF = (Tf * IDF)$$

Sentence	good (feature 1)	boy (feature 2)	girl (feature 3)
Sentence_1	0	$(\frac{1}{2} * \log\frac{3}{2})$	0
Sentence_2	0	0	$(\frac{1}{2} * \log\frac{3}{2})$
Sentence_3	0	$(\frac{1}{3} * \log\frac{3}{2})$	$(\frac{1}{3} * \log\frac{3}{2})$

Term Frequency- Inverse Document Frequency (Tf-IDF)

Advantages:

- With Tf-IDF **Semantic Meaning is Considered**:
  - Sentence_1 : Talks about the Boy and it is given Weightage
  - Sentence_2 : Talks about the Girl and it is given Weightage
  - Sentence_3: Talks about both Boy and Girl and so both are given Weightage

Sentence	good (feature 1)	boy (feature 2)	girl (feature 3)
Sentence_1	0	$(\frac{1}{2} * \log\frac{3}{2})$	0
Sentence_2	0	0	$(\frac{1}{2} * \log\frac{3}{2})$
Sentence_3	0	$(\frac{1}{3} * \log\frac{3}{2})$	$(\frac{1}{3} * \log\frac{3}{2})$

Syntactic Level

- Vector-space model
- **Language models**
- Full-parsing
- Cross-modality

Language Models

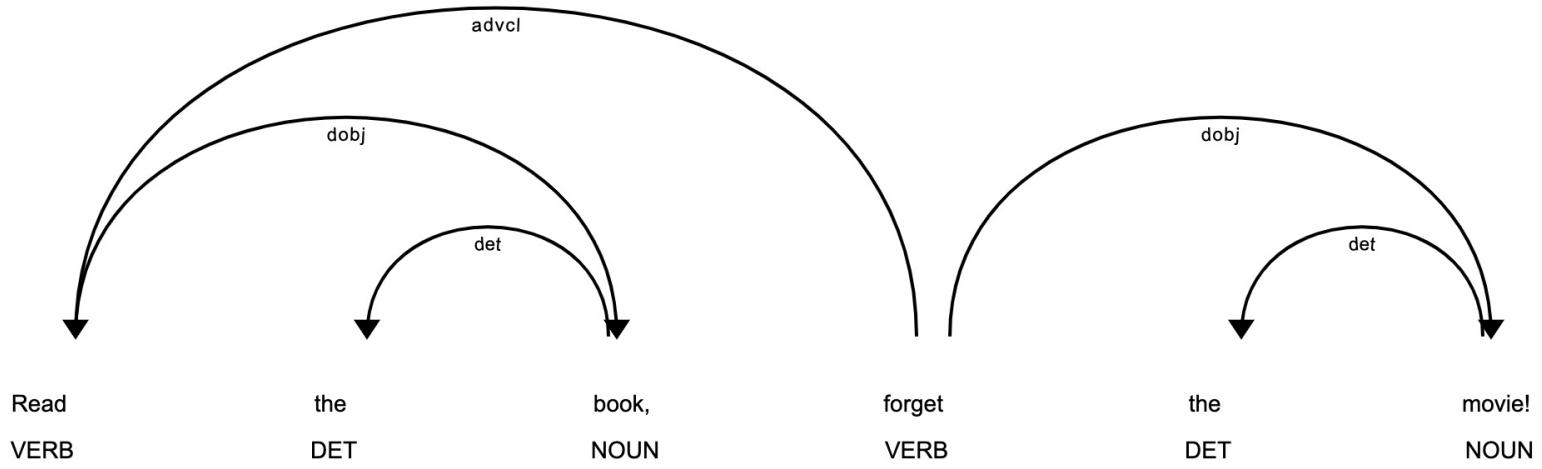
- Language modeling is about determining probability of a sequence of words
  - Task typically gets reduced to the estimating probabilities of a next word given two previous words (Trigram model)
  - Applications including speech recognition, OCR, handwriting recognition, machine translation and spelling correction

Syntactic Level

- Vector-space model
- Language models
- **Full-parsing**
- Cross-modality

Full-parsing

- Parsing of a text provides the **maximum structural information** of sentences
- **Input** we provide a **sentence** and **output** is a **tree**
- For most of the methods dealing with the text data the information in parse trees is too complex.



Syntactic Level

- Vector-space model
- Language models
- Full-parsing
- **Cross-modality**

Cross-modality Level

- Often the cases that objects are represented with different data types:
  - Text documents
  - Multilingual texts documents
  - Images
  - Videos
  - Social Networks
  - Sensor Networks
- How to create mapping between different representation so that we can benefit using more information about the same objects

Semantic Level

Semantic Level

- **Collaborative tagging**
- Templates
- Ontologies

Collaborative tagging

- Collaborative tagging is a process of adding metadata to annotate content (example: documents, web sites, photos)
  - Metadata is typically a form of keywords
  - It can be done manually by different users and as a result annotated data

- Tags to the photos on:
  - https://www.flickr.com/photos/15631880@N08/51135082081/in/gallery-flickr-72157719057763396/

Semantic Level

- Collaborative tagging
- **Templates**
- Ontologies

Templates

- Templates are the mechanism for extracting the information from text
  - Focused on specific domain that includes consistent patterns on where specific information is positioned
  - Basic methods for information extraction

- Generic Templates:
  - <NP> “is a” <class>
  - <class> “such as” <NP>

- Each template represents specific relationship between the words appearing in the variable slots

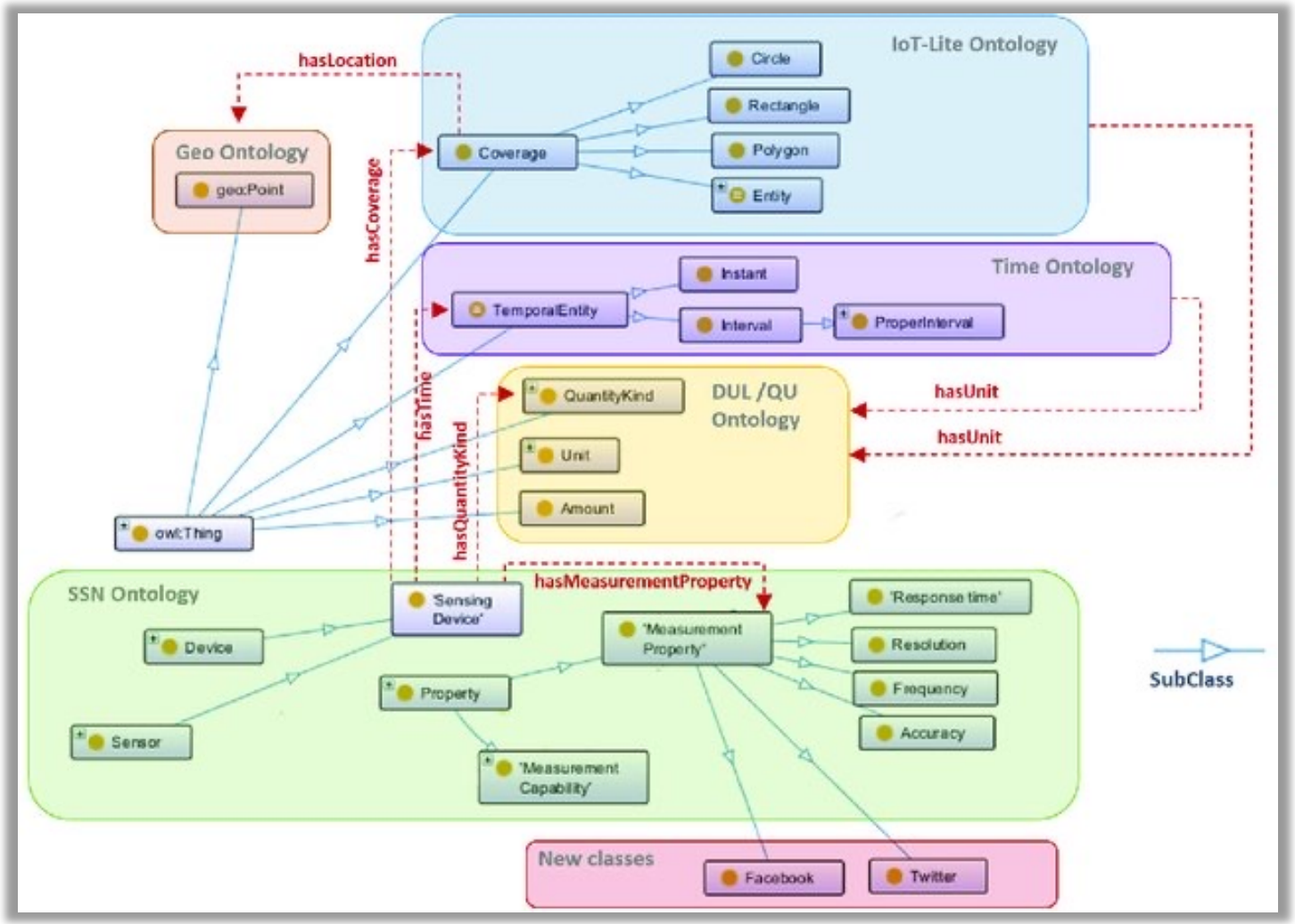
Semantic Level

- Collaborative tagging
- Templates
- **Ontologies**

Ontologies

- Ontologies are the most general formalism for describing data objects
  - Ontologies are most popular in Semantic Web and OWL standards
  - Ontologies can be of various complexity – from simple ones (light weight described with simple) to heavy weight (described with first order theories)
  - Ontologies could be understood also as very generic data-models where we can store extracted information from text

Ontologies



[Valtolina et al. 2019]

Dealing with Textual data

02

Advantages of dealing with Textual data

— Stock Price Prediction

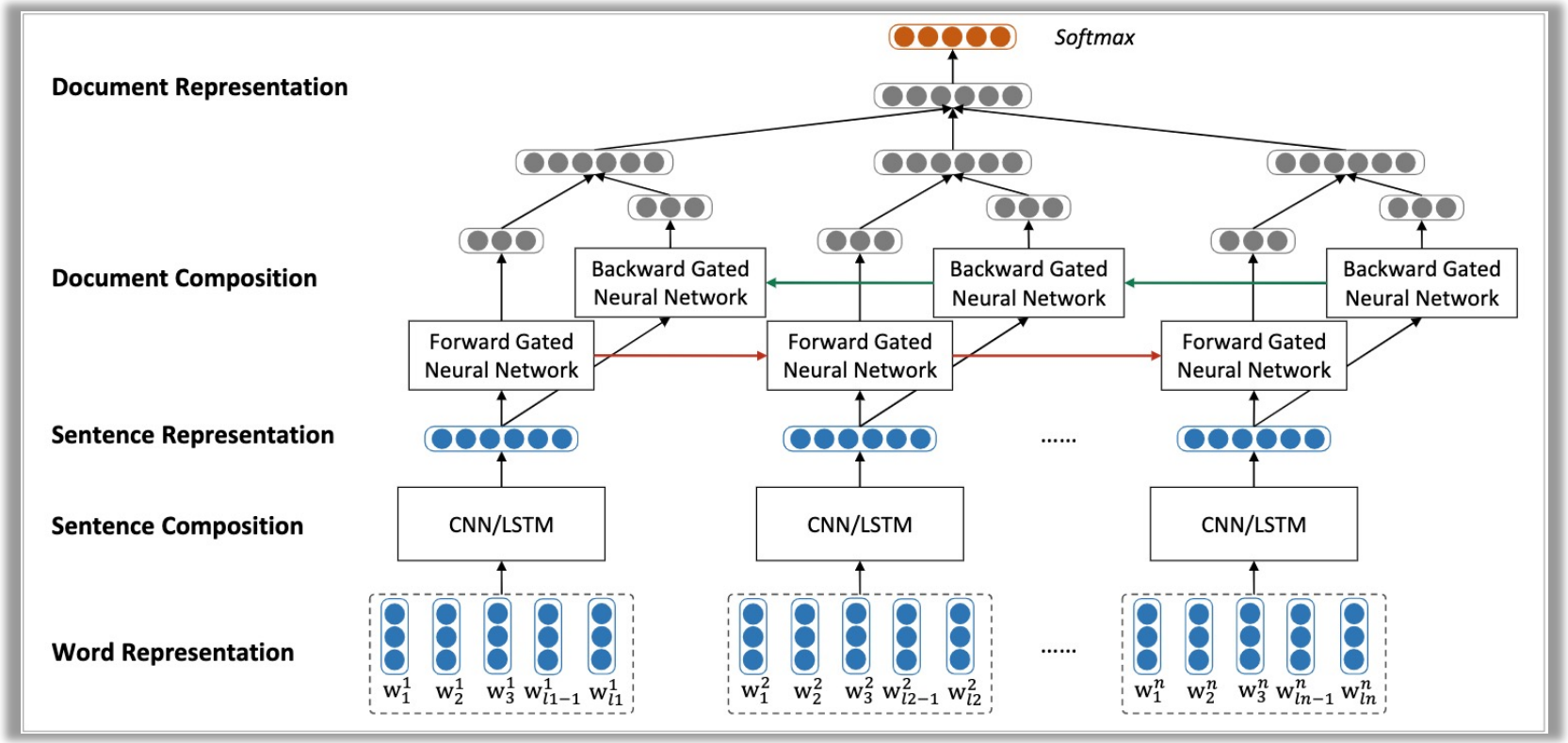
Selvin, S., Vinayakumar, R., Gopalakrishnan, E.A., Menon, V.K. and Soman, K.P., 2017, September. **Stock price prediction using LSTM, RNN and CNN-sliding window model.** In *2017 international conference on advances in computing, communications and informatics (icacci)* (pp. 1643-1647). IEEE.

— Voice Assistant

Image source: https://www.softwaretestinghelp.com/voice-recognition-software

Advantages of dealing with Textual data

— Document level sentiment classification



Tang, D., Qin, B. and Liu, T., 2015, September. **Document modelling with gated recurrent neural network for sentiment classification.** In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1422-1432).

Advantages of dealing with Textual data

— Named Entity Recognition System

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K. and Dyer, C., 2016. **Neural architectures for named entity recognition.** *arXiv preprint arXiv:1603.01360*.

— Part of Speech Tagging System

Wang, P., Qian, Y., Soong, F.K., He, L. and Zhao, H., 2015. **Part-of-speech tagging with bidirectional long short-term memory recurrent neural network.** *arXiv preprint arXiv:1510.06168*.

Advantages of dealing with Textual data

— Text Summarization

Gold Summary: Redpath has ended his eight-year association with Sale Sharks. Redpath spent five years as a player and three as a coach at sale. He has thanked the owners, coaches and players for their support.	Salience	Content	Novelty	Position	Prob.
Bryan Redpath has left his coaching role at Sale Sharks with immediate effect.	0.1	0.1	0.9	0.1	0.3
The 43 - year - old Scot ends an eight-year association with the Aviva Premiership side, having spent five years with them as a player and three as a coach.	0.9	0.6	0.9	0.9	0.7
Redpath returned to Sale in June 2012 as director of rugby after starting a coaching career at Gloucester and progressing to the top job at Kingsholm .	0.8	0.5	0.5	0.9	0.6
Redpath spent five years with Sale Sharks as a player and a further three as a coach but with Sale Sharks struggling four months into Redpath's tenure, he was removed from the director of rugby role at the Salford-based side and has since been operating as head coach .	0.8	0.9	0.7	0.8	**0.9**
'I would like to thank the owners, coaches, players and staff for all their help and support since I returned to the club in 2012.	0.4	0.1	0.1	0.7	0.2
Also to the supporters who have been great with me both as a player and as a coach,' Redpath said.	0.6	0.0	0.2	0.3	0.2

Nallapati, R., Zhai, F. and Zhou, B., 2017, February. **Summarunner: A recurrent neural network based sequence model for extractive summarization of documents.** In *Thirty-First AAAI Conference on Artificial Intelligence*.

Advantages of dealing with Textual data

— Image Captioning

A baby with a toothbrush in its mouth

Karpathy, A. and Fei-Fei, L., 2015. **Deep visual-semantic alignments for generating image descriptions.** In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3128-3137).

You, Q., Jin, H., Wang, Z., Fang, C. and Luo, J., 2016. **Image captioning with semantic attention.** In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4651-4659).

Advantages of dealing with Textual data

— Video Captioning

Wang, B., Ma, L., Zhang, W. and Liu, W., 2018. **Reconstruction network for video captioning.** In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 7622-7631).

Advantages of dealing with Textual data

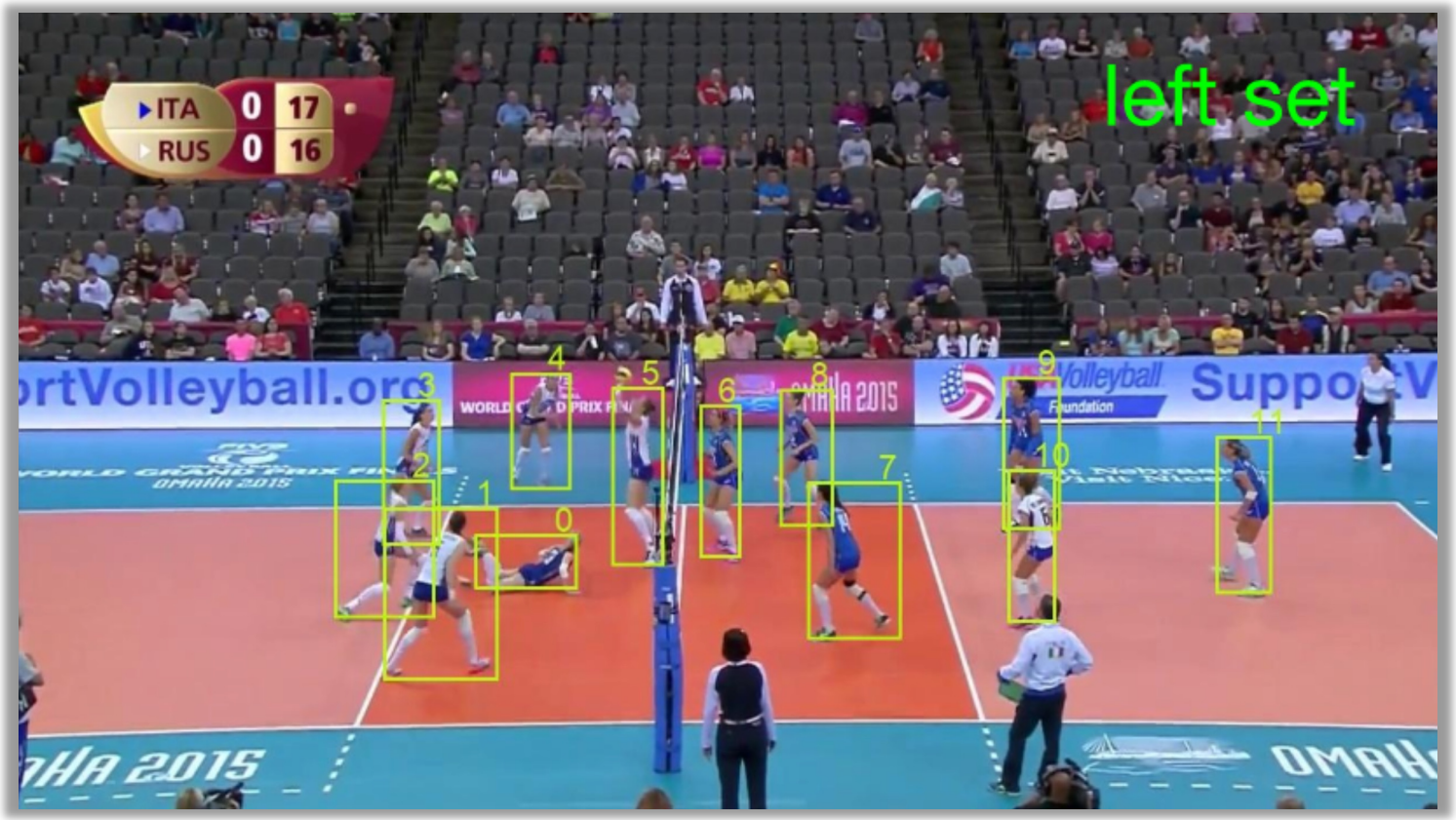
— Image Generation

Generated MNIST images with two digits

Gregor, K., Danihelka, I., Graves, A., Rezende, D.J. and Wierstra, D., 2015. **Draw: A recurrent neural network for image generation.** *arXiv preprint arXiv:1502.04623*.

Advantages of dealing with Textual data

— Group Activity Recognition (Transformer Model)



Gavrilyuk, K., Sanford, R., Javan, M. and Snoek, C.G., 2020. **Actor-Transformers for Group Activity Recognition.** *arXiv preprint arXiv:2003.12737.*

What is Textual data?

What is Textual data?

Sequence Data

According to Mutz [8], social media communication also has a significant impact on the content and form of sports reporting. The term social media stands for the exchange of information, experiences and opinions through community websites, that leads to growing complexity in communication about the number and heterogeneity of individuals involved in communication and the degree of diffusion of information about reach, scale and speed [9] [10]. The process of generating and sharing information about a sports event is particularly interesting from a research point of view since a small group of institutionalised ”gatekeepers” [11] no longer control the diffusion of publicly available information. They are thus allowing the assumption that significantly fewer neutral statements are disseminated there than in the established media [12]. The research project [13] address the above issue by showing to what extent the media discourse in social media in Germany contributes to thematising, constructing and staging national identity and that identity motives become visible in the online communication. The research project aims to examine the content of use of Twitter communication during the period of the UEFA European Championships in 2016 concerning German matches.

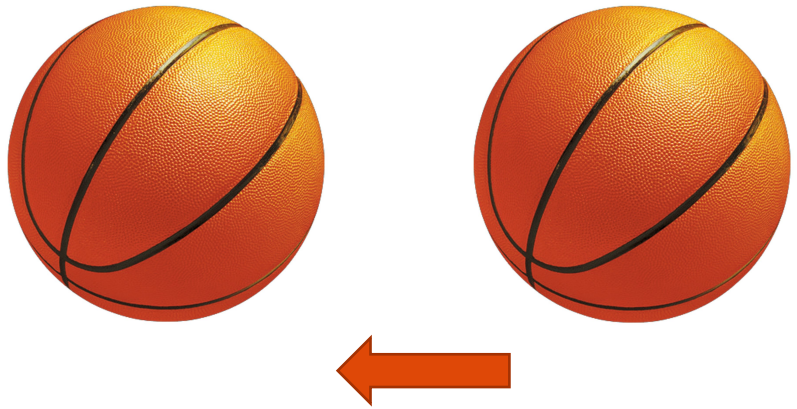
What is Sequence data?

What is Sequence data?

— Snapshot of ball in Motion

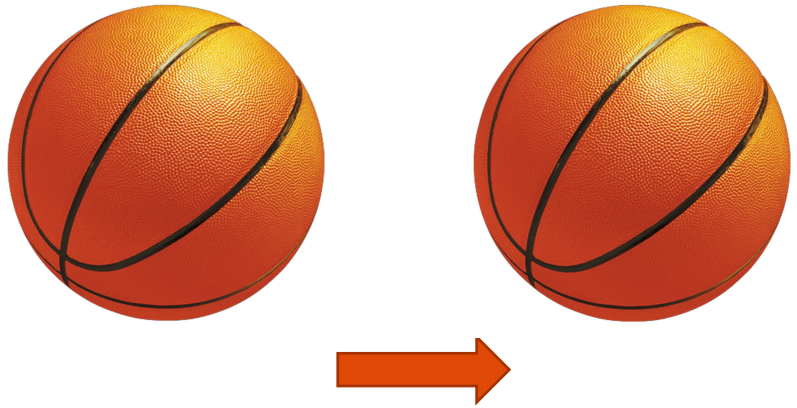
What is Sequence data?

- Snapshot of ball in Motion
- Can you predict the direction of ball in motion?



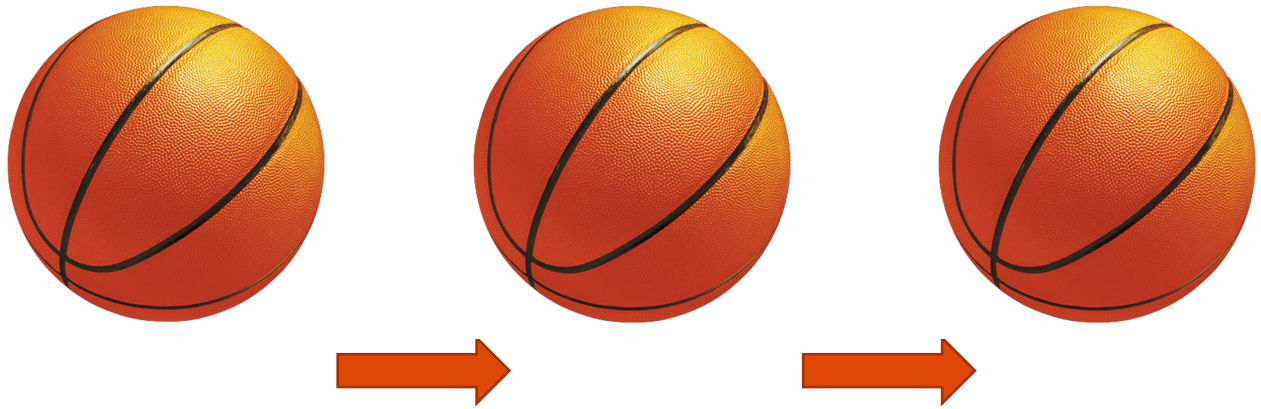
What is Sequence data?

- Snapshot of ball in Motion
- Can you predict the direction of ball in motion?



What is Sequence data?

- Snapshot of ball in Motion
- Can you predict the direction of ball in motion?



Various forms of Sequence data

— Text

According to Mutz [8], social media communication also has a significant impact on the content and form of sports reporting. The term social media stands for the exchange of information, experiences and opinions through community websites, that leads to growing complexity in communication about the number and heterogeneity of individuals involved in communication and the degree of diffusion of information about reach, scale and speed [9] [10]. The process of generating and sharing information about a sports event is particularly interesting from a research point of view since a small group of institutionalised ”gatekeepers” [11] no longer control the diffusion of publicly available information. They are thus allowing the assumption that significantly fewer neutral statements are disseminated there than in the established media [12]. The research project [13] address the above issue by showing to what extent the media discourse in social media in Germany contributes to thematising, constructing and staging national identity and that identity motives become visible in the online communication. The research project aims to examine the content of use of Twitter communication during the period of the UEFA European Championships in 2016 concerning German matches.

Various forms of Sequence data

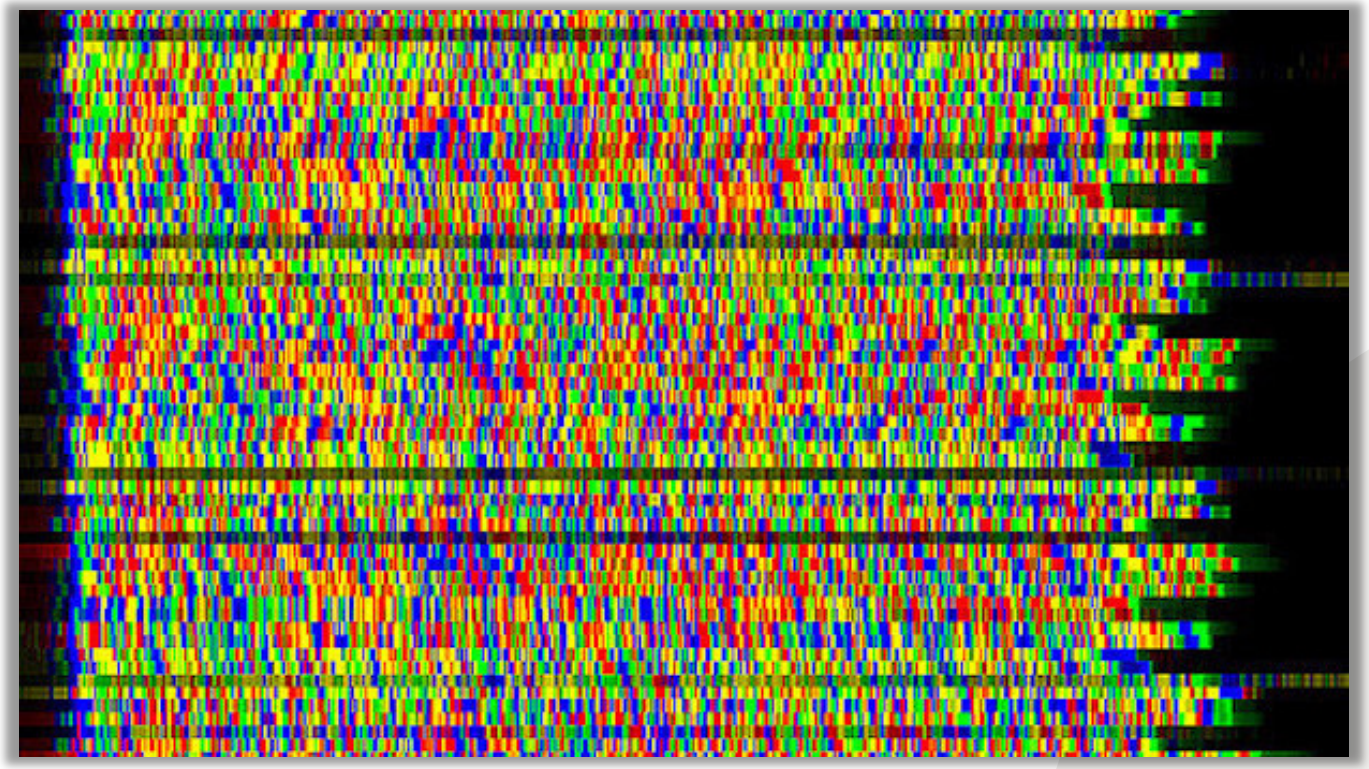
- Text
- Time series

Various forms of Sequence data

- Text
- Time series
- Videos

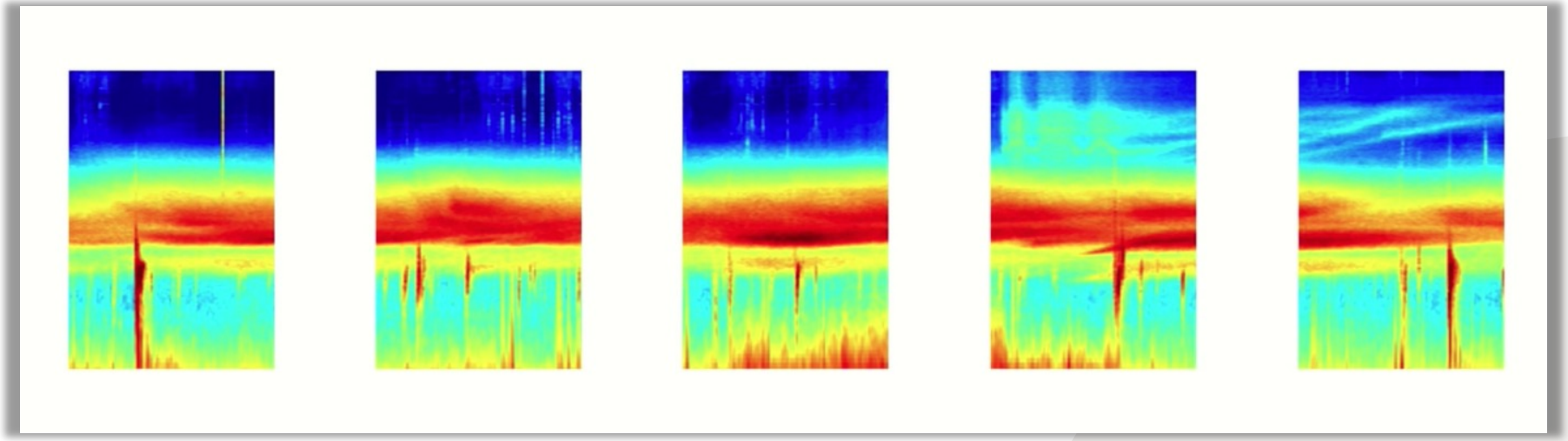
Various forms of Sequence data

- Text
- Time series
- Videos
- DNA sequences



Various forms of Sequence data

- Text
- Time series
- Videos
- DNA sequences
- Audio



How Humans process Sequential data?

How Humans process Sequential data?

Humans can predict the context of the text provided based on previous understanding due to a "Human Memory" or "Retention Power" or "Sequential Memory".

Sequential memory

Sequential memory

"working love learning we on deep" : **Does it make sense?**

"We love working on deep learning" : **Make sense**

Sequence of words define their meaning. If such sequence of information data determining the event is used to predict the output, then a network having access to **prior knowledge is required**.

Sequential memory

— Say alphabets in your head from A-Z

Sequential memory

— Say alphabets in your head from A-Z

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Sequential memory

- Say alphabets in your head from A-Z

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

- Say the alphabets A-Z now in the reverse order

Sequential memory

- Say alphabets in your head from A-Z

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

- Say the alphabets A-Z now in the reverse order

Z Y X W V U T S R Q P O N M L K J I H G F E D C B A

Sequential memory

— Middle of a verse

You are my fire

The one desire

You are

You are, you are, you are

Don't wanna hear you say

Ain't nothing but a heartache

Ain't nothing but a mistake (don't wanna hear you say)

Sequential memory

— Song

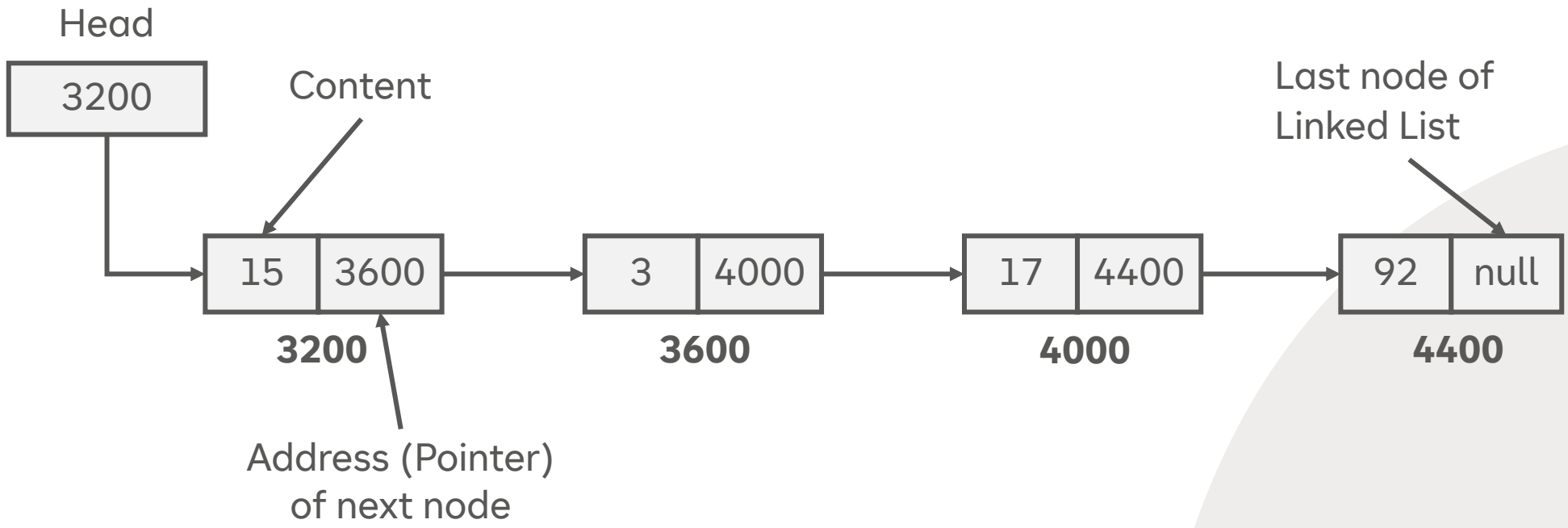
"I Want It That Way" - Backstreet Boys

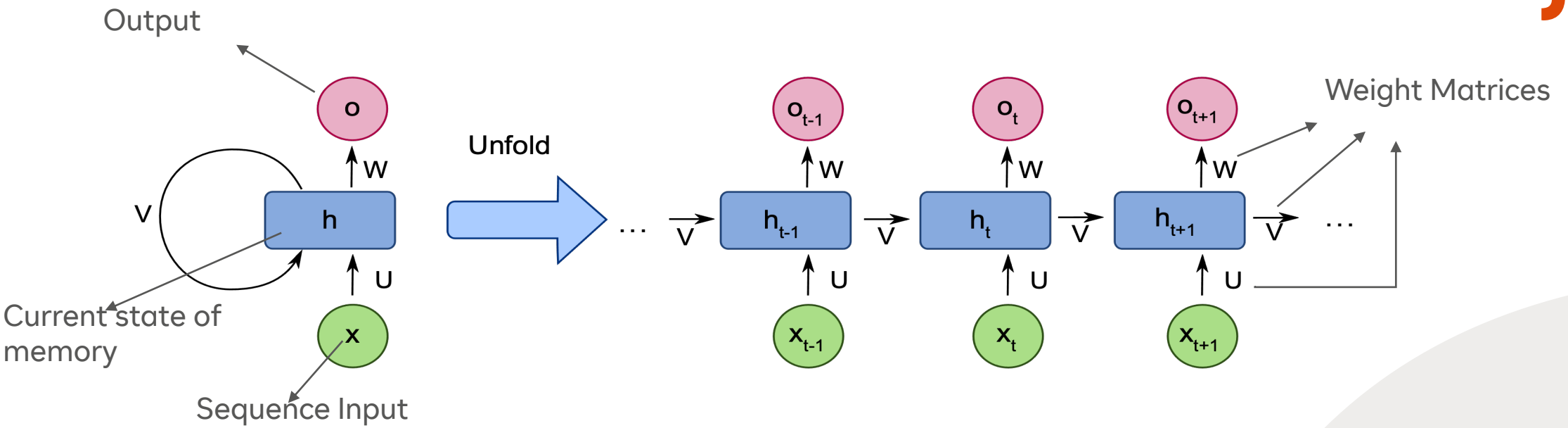
Sequential memory

- Alphabets are not learned in reverse manner.
- Songs are not learned in backward fashion.

“Everything is learned in SEQUENCE”

Linked list


Head
3200
Content
15
3600
3
4000
17
4400
92
null
Last node of
Linked List
3200
3600
4000
4400
Address (Pointer)
of next node

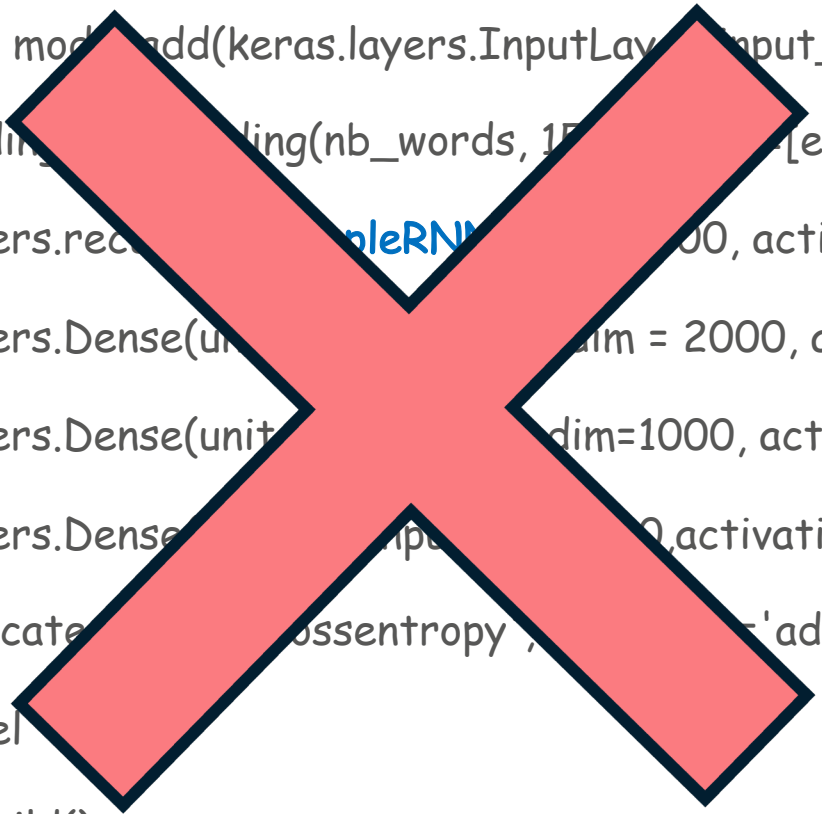


Recurrent Neural Network

What is Recurrent Neural Network?

1. model = Sequential() model.add(keras.layers.InputLayer(input_shape=(15,1)))
2. keras.layers.embeddings.Embedding(nb_words, 15, weights=[embedding_matrix], input_length=15, trainable=False)
3. model.add(keras.layers.recurrent.SimpleRNN(units = 100, activation='relu', use_bias=True))
4. model.add(keras.layers.Dense(units=1000, input_dim = 2000, activation='sigmoid'))
5. model.add(keras.layers.Dense(units=500, input_dim=1000, activation='relu'))
6. model.add(keras.layers.Dense(units=2, input_dim=500,activation='softmax'))
7. model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
8. #compiling the model
9. finalmodel = modelbuild()

What is Recurrent Neural Network?



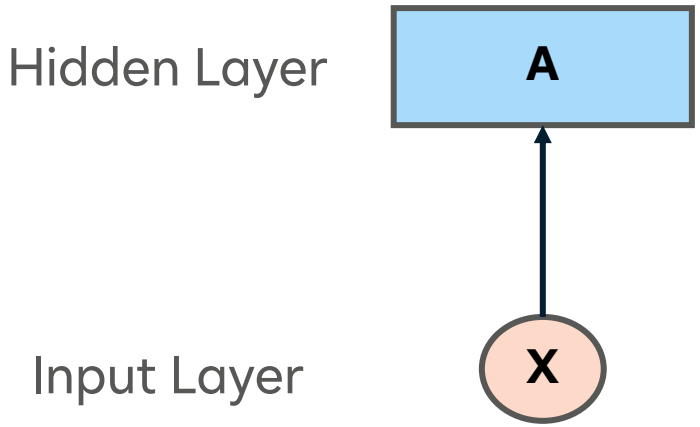
1. model = Sequential() mod add(keras.layers.InputLay nput_shape=(15,1)))
2. keras.layers.embeddi ling(nb_words, 1 [embedding_matrix], input_length=15, trainable=False)
3. model.add(keras.layers.rec pleRN 00, activation='relu', use_bias=True))
4. model.add(keras.layers.Dense(u im = 2000, activation='sigmoid'))
5. model.add(keras.layers.Dense(unit dim=1000, activation='relu'))
6. model.add(keras.layers.Dense inp 0,activation='softmax'))
7. model.compile(loss='cate ossentropy , 'adam', metrics=['accuracy'])
8. #compiling the model
9. finalmodel = modelbuild()

How traditional Neural Network work?

— Traditional feed-forward neural network.

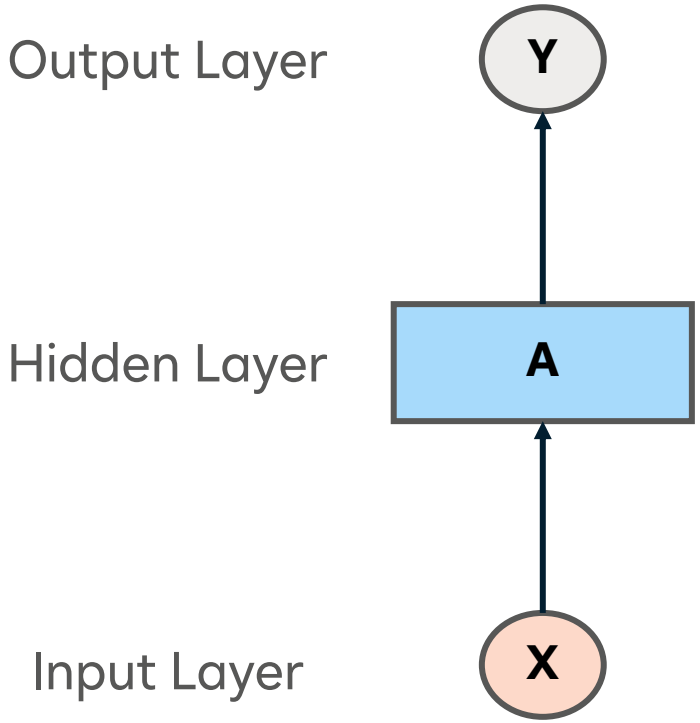
How traditional Neural Network work?

— Traditional feed-forward neural network.



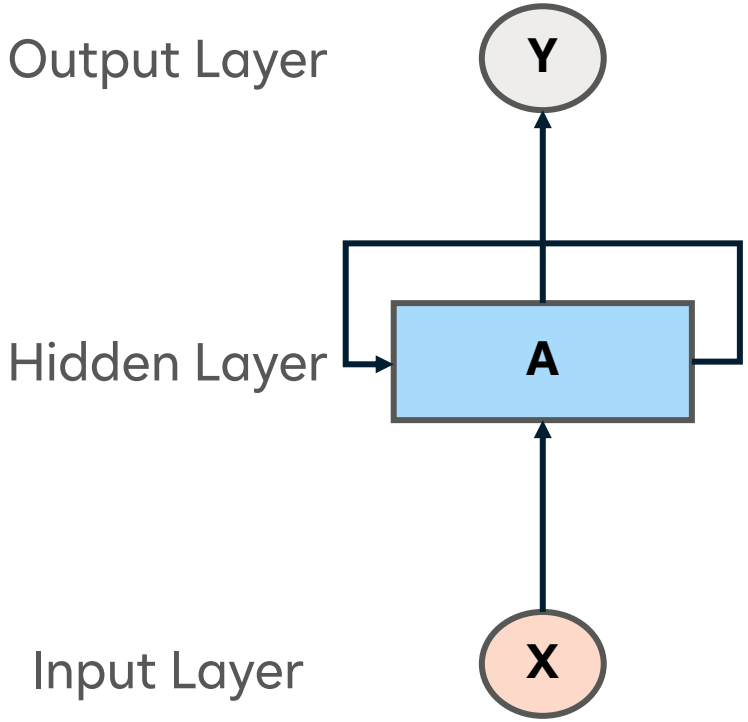
How traditional Neural Network work?

— Traditional feed-forward neural network.



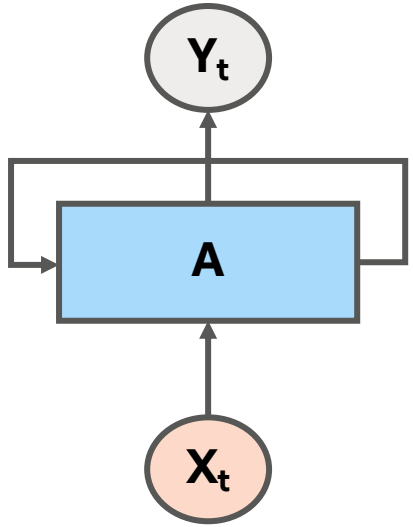
What is Recurrent Neural Network?

- Loop in the traditional feed forward neural network.
- RNN is having the concept of sequential memory.



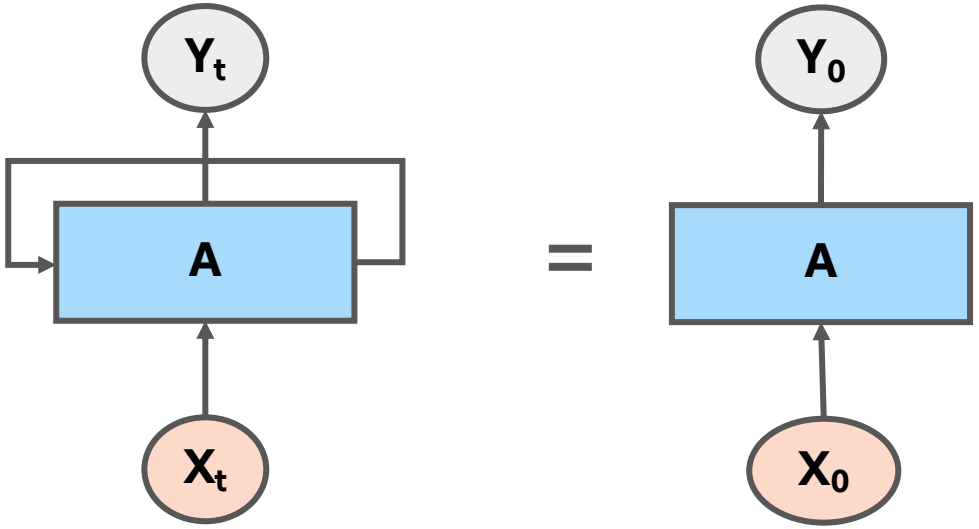
What is Recurrent Neural Network?

- RNN is a sequence of neural network blocks that are linked in the form of chain.
- Each block is passing message to the successor block.



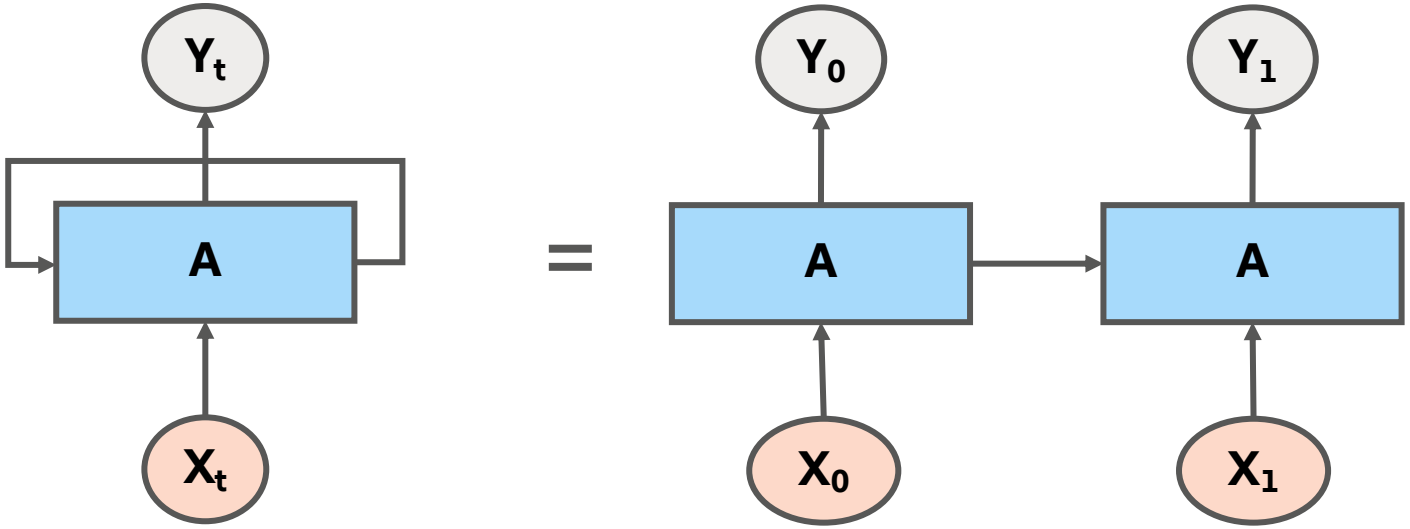
What is Recurrent Neural Network?

- RNN is a sequence of neural network blocks that are linked in the form of chain.
- Each block is passing message to the successor block.



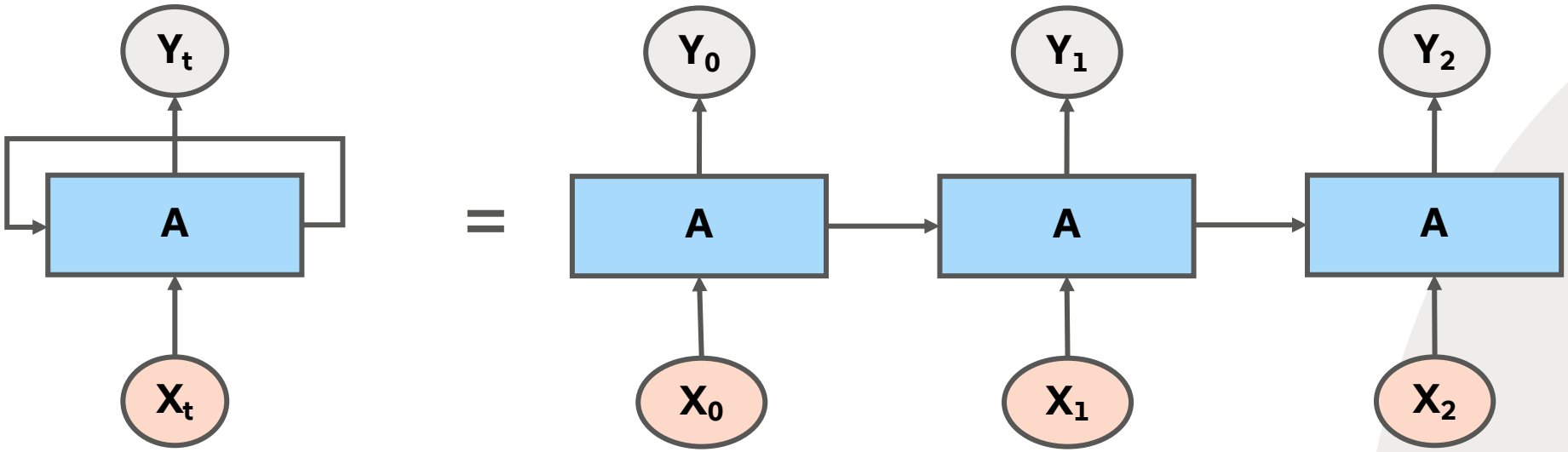
What is Recurrent Neural Network?

- RNN is a sequence of neural network blocks that are linked in the form of chain.
- Each block is passing message to the successor block.



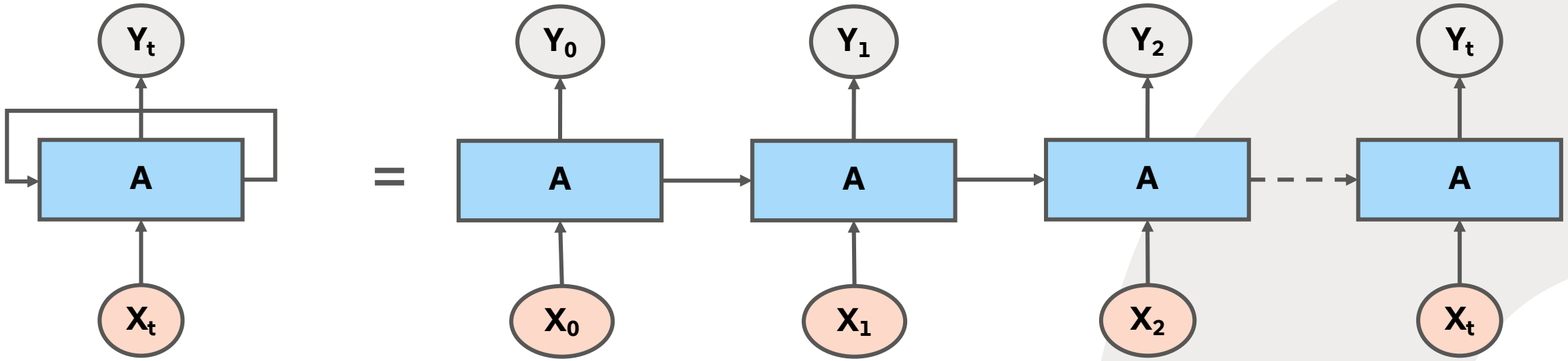
What is Recurrent Neural Network?

- RNN is a sequence of neural network blocks that are linked in the form of chain.
- Each block is passing message to the successor block.



What is Recurrent Neural Network?

- RNN is a sequence of neural network blocks that are linked in the form of chain.
- Each block is passing message to the successor block.



What is Recurrent Neural Network?

Recurrent Neural Network (RNN) can be used for mapping inputs to outputs of varying types, lengths and are fairly generalized in their application.

How Recurrent Neural Network Work?

- Supplying Input to the hidden layer.
- Weights and biases of the recurrent neuron would be the same (weight and biases of all hidden layer (memory) are same).
- Recurrent neuron stores the previous input information and combines with the current input thereby preserving some relationship of the current input with the previous input.

How Recurrent Neural Network Work?

(input + empty_memory) -> memory -> output

(input + empty_input) -> memory -> output

How Recurrent Neural Network Work?

(input + empty_memory) -> memory -> output

(input + prev_memory) -> memory -> output

(input + empty_input) -> memory -> output

(input + prev_input) -> memory -> output

How Recurrent Neural Network Work?

(input + empty_memory) -> memory -> output

(input + prev_memory) -> memory -> output

(input + prev_ memory) -> memory -> output

(input + empty_input) -> memory -> output

(input + prev_input) -> memory -> output

(input + prev_input) -> memory -> output

How Recurrent Neural Network Work?

(input + empty_memory) -> memory -> output

(input + prev_memory) -> memory -> output

(input + prev_ memory) -> memory -> output

(input + prev_memory) -> memory -> output

(input + empty_input) -> memory -> output

(input + prev_input) -> memory -> output

(input + prev_input) -> memory -> output

(input + prev_input) -> memory -> output

How Recurrent Neural Network Work?

(input + empty_memory) -> memory -> output

(input + prev_memory) -> memory -> output

(input + prev_ memory) -> memory -> output

(input + prev_memory) -> memory -> output

(input + empty_input) -> memory -> output

(input + prev_input) -> memory -> output

(input + prev_input) -> memory -> output

(input + prev_input) -> memory -> output

Recurrent Neural Network Memory

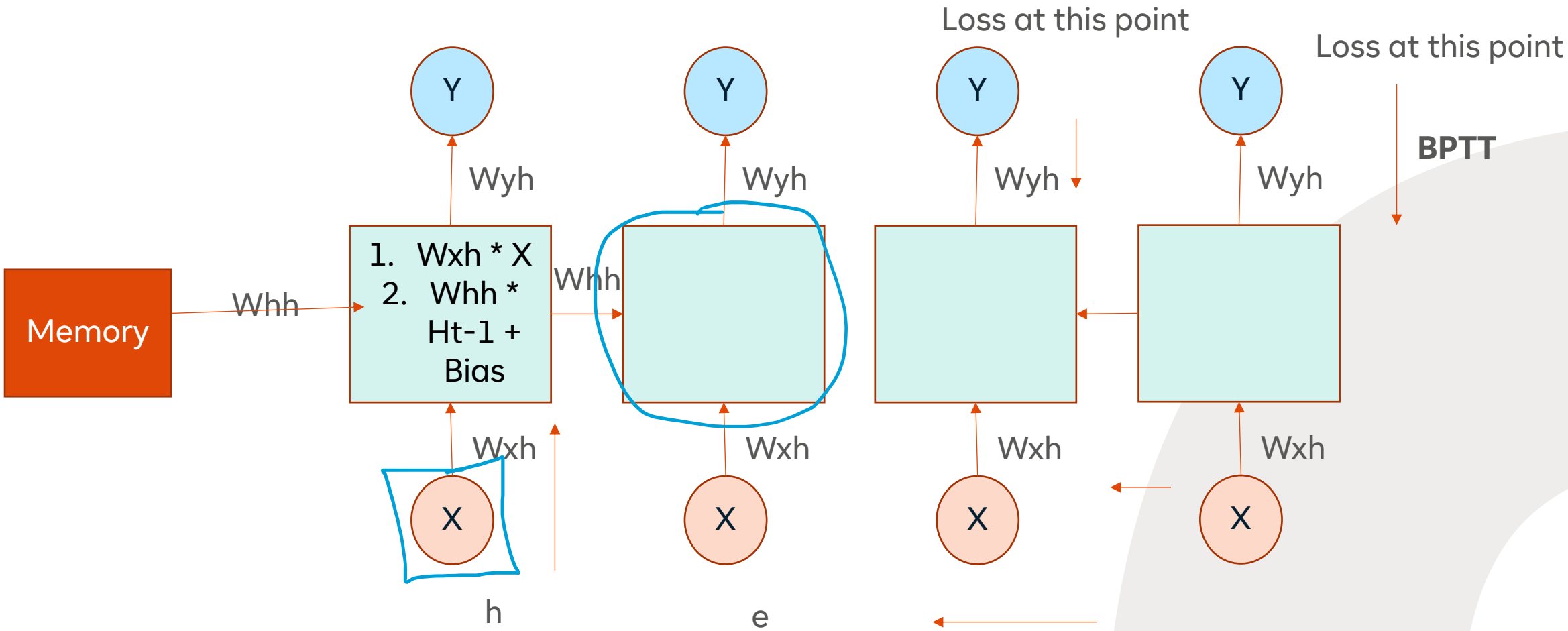
- https://github.com/myaooo/RNNVis
- https://www.myaooo.com/projects/rnnvis/

Demo

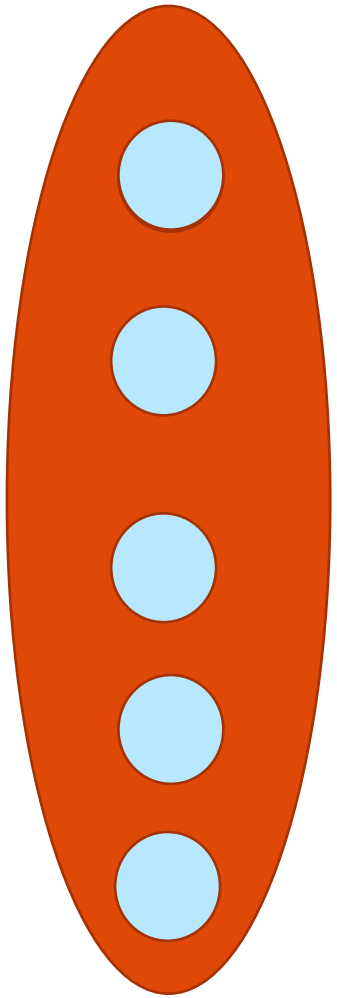
Character level prediction

Hell_: Hello Hello, Who, ... Dictionary: {h, e, l, o, w)

Synapse_0 : Wxh: Input is 2 , memory : 16
Synapse_1: Wyh: memory: 16 and output: 1
Synapse_h: Whh: 16 * 16

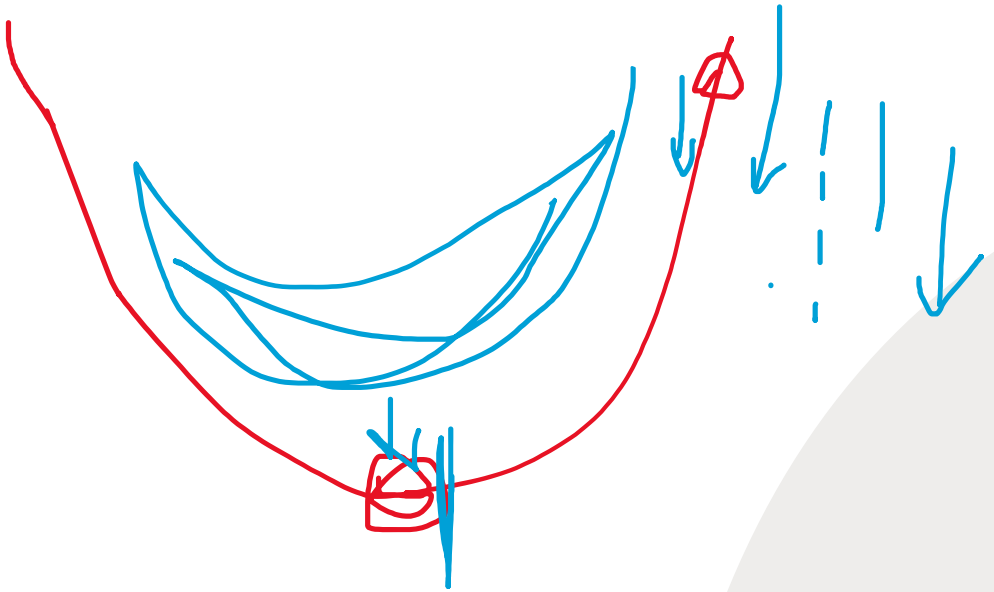


Binary Addition



9: 00001001
60: 00111100
:
69: 0100101

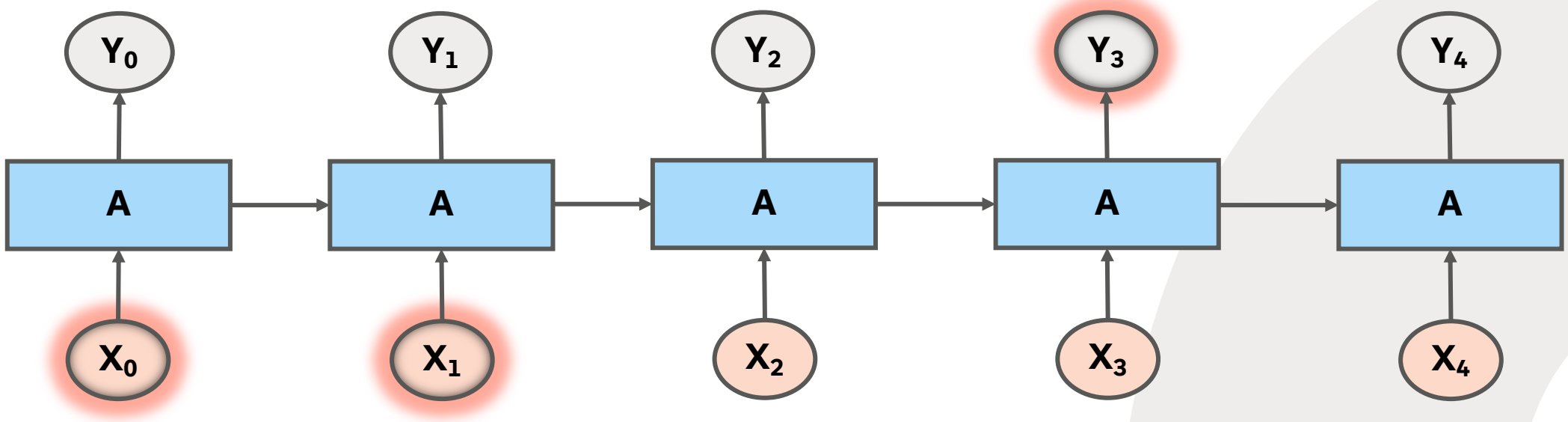
(1,1) -> 0 (1 carry over)
(0,0)



Drawbacks of Recurrent Neural Network

- RNN is able to connect the previous information to the present tasks.
- Example:

  "The Clouds are in the ____." => sky



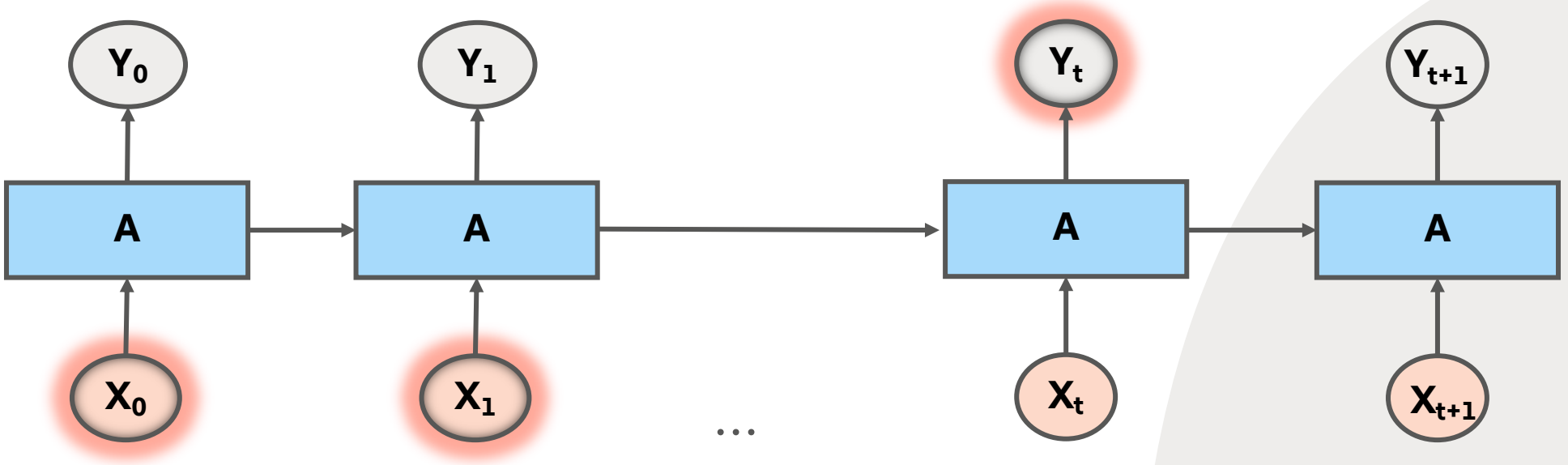
Drawbacks of Recurrent Neural Network

— Problem:

"I grew up in France…… I speak fluent _____." => French

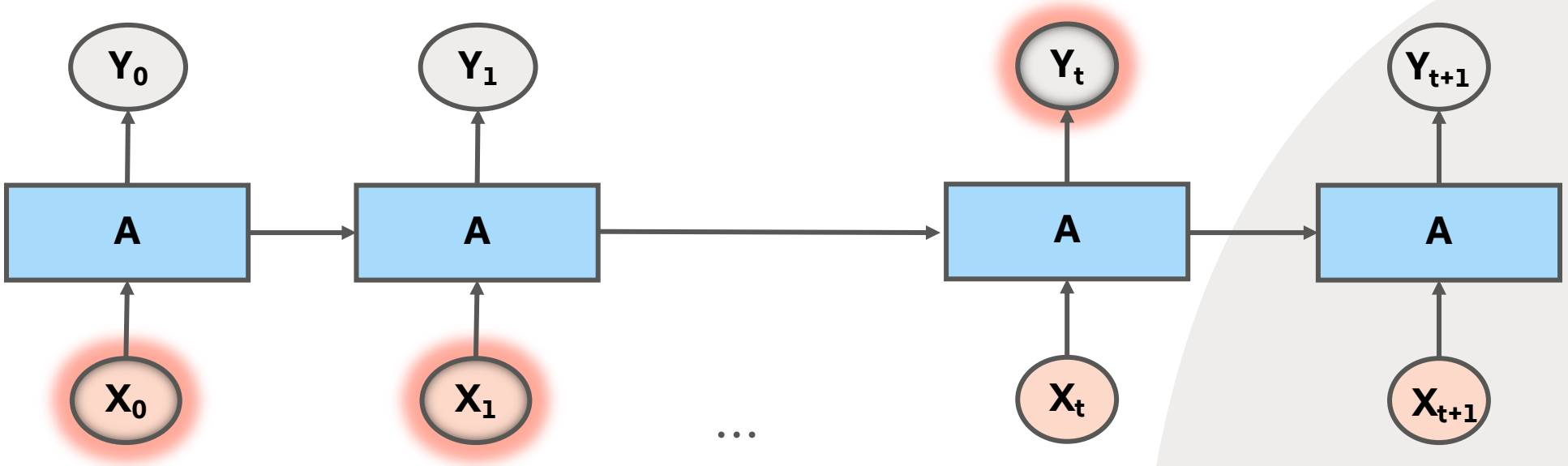
RNN can fill the blank with a word => Name of the language.

Narrow down the language => France word is relevant in the statement.



Drawbacks of Recurrent Neural Network

- Bengio, Y., Simard, P. and Frasconi, P., 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, *5*(2), pp.157-166.
- Vanishing and Exploding Gradient problem.



Solutions for Recurrent Neural Network

- Long Short Term Memory (LSTM) are kind of RNN that are capable of learning long-term dependencies.
- LSTM are introduced by Hochreiter and Schmidhuber (1997)

  [Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.]
- Attention in LSTM

  [Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R. and Bengio, Y., 2015, June. Show, attend and tell: Neural image caption generation with visual attention. In International conference on machine learning (pp. 2048-2057).]

Solutions for Recurrent Neural Network

- Grid LSTM

  [Kalchbrenner, N., Danihelka, I. and Graves, A., 2015. Grid long short-term memory. arXiv preprint arXiv:1507.01526.]

- RNN in generative models

  [Gregor, K., Danihelka, I., Graves, A., Rezende, D.J. and Wierstra, D., 2015. Draw: A recurrent neural network for image generation. arXiv preprint arXiv:1502.04623.]

References

1. Han, J. and Moraga, C., 1995, June. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks* (pp. 195-201). Springer, Berlin, Heidelberg.
2. Hecht-Nielsen, R., 1992. Theory of the backpropagation neural network. In *Neural networks for perception* (pp. 65-93). Academic Press.
3. Zaremba W, Sutskever I, Vinyals O. Recurrent neural network regularization. arXiv preprint arXiv:1409.2329. 2014 Sep 8.
4. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. and Kuksa, P., 2011. Natural language processing (almost) from scratch. Journal of machine learning research, 12(Aug), pp.2493-2537.
5. Kim, Y., 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.
6. Chen, G., 2016. A gentle tutorial of recurrent neural network with error backpropagation. *arXiv preprint arXiv:1610.02583*.
7. Hearst, Marti A. "Text Tiling: Segmenting text into multi-paragraph subtopic passages." *Computational linguistics* 23, no. 1 (1997): 33-64.

References

8. Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. "Introduction to WordNet: An on-line lexical database." *International journal of lexicography* 3, no. 4 (1990): 235-244.

9. Kamps, Jaap, and Maarten Marx. "Visualizing wordnet structure." In *Proc. of the 1st International Conference on Global WordNet*, pp. 182-186. 2002.

10. Tutorial on Text Mining and Link Analysis for Web and Semantic Web by Marko Grobelnik and Dunja Mladenic in 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Jose 2007.

11. https://www.ldc.upenn.edu/about

12. Valtolina, Stefano, Luca Ferrari, and Marco Mesiti. "Ontology-Based Consistent Specification of Sensor Data Acquisition Plans in Cross-Domain IoT Platforms." *IEEE Access* 7 (2019): 176141-176169.

Appendix

— Sigmoid function and derivative of sigmoid function [1]:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$\frac{d\sigma(x)}{dx} = \frac{d}{dx}\frac{1}{(1 + e^{-x})}$$

Apply reciprocal rule: $\frac{df(x)}{dx} = \frac{d}{dx}\frac{1}{f(x)} = -\frac{f'(x)}{f(x)^2}$

Apply rule of linearity: $[a\,.\,u(x) + b\,.\,v(x)]' = a\,.\,u'(x) + b\,.\,v'(x)$

Exponential rule: $\left[e^{u(x)}\right]' = e^{u(x)}\,.\,u'(x)$

Appendix

$$\frac{d\sigma(x)}{dx} = \frac{e^{-x}}{(1+e^{-x})^2}$$

In order to simplify the equation, we can write the above equation as:

$$\frac{d\sigma(x)}{dx} = \frac{1\,.\,e^{-x}}{(1+e^{-x})\,.\,(1+e^{-x})}$$

$$\frac{d\sigma(x)}{dx} = \frac{1}{(1+e^{-x})} \cdot \frac{e^{-x}+1\ -1}{(1+e^{-x})}$$

$$\frac{d\sigma(x)}{dx} = (\sigma(x)\,.\,(1-\sigma(x)))$$

Appendix

— Tanh function and derivative of tanh function:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

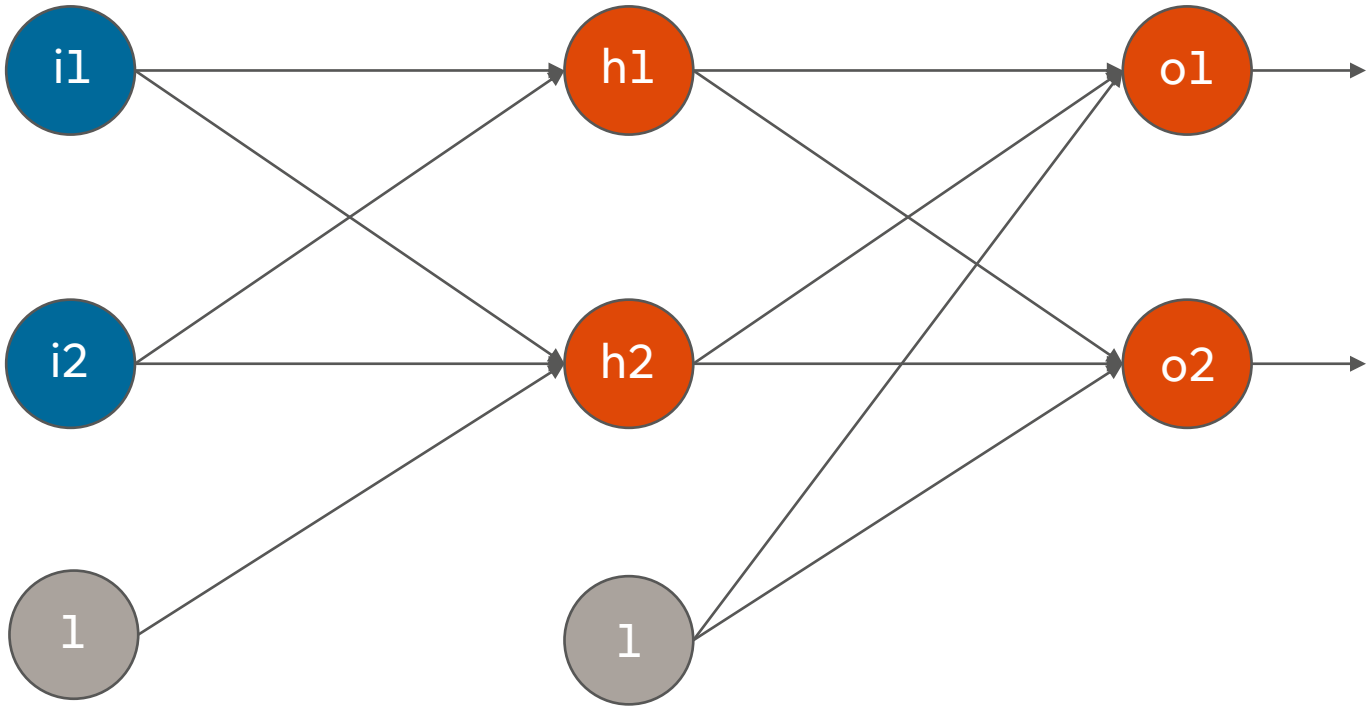
$$\frac{dtanh(x)}{dx} = 1 - \frac{(e^x - e^{-x})^2}{(e^x + e^{-x})^2} = 1 - tanh^2(x)$$

— Softmax function:

$$\text{Softmax}(x_j) = \frac{e^{x_j}}{\sum_{k=1}^{n} e^{x_k}}$$

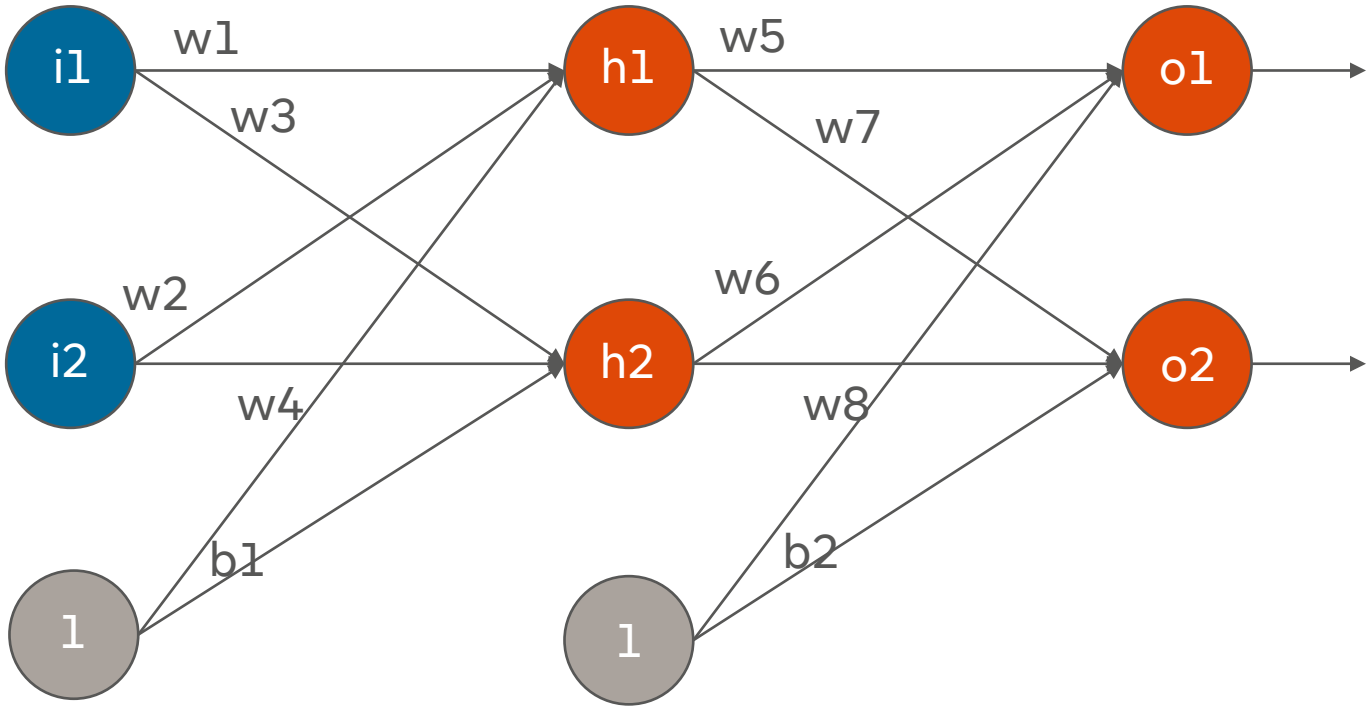
Appendix

— Back Propagation in neural network [2]



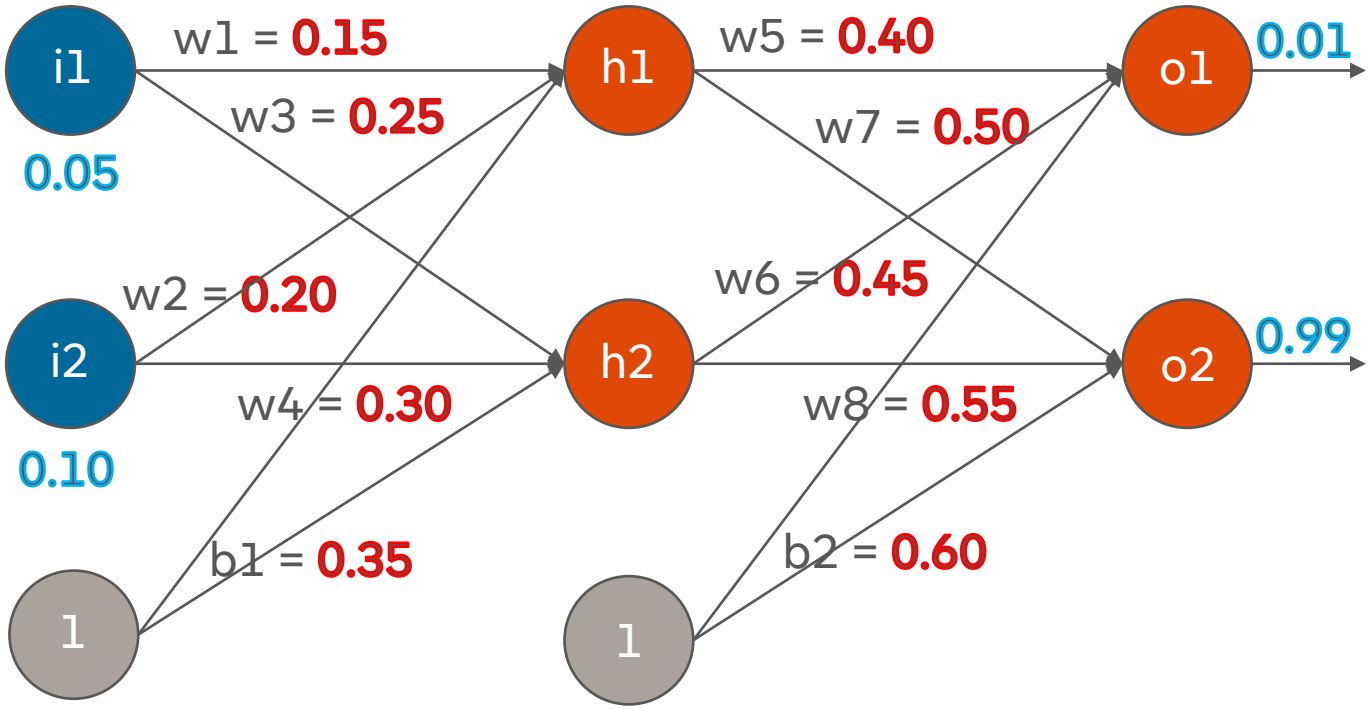
Appendix

— Weights are assigned to all the connection in the neural network.



Appendix

— A random weight value is assigned represented in red, and random input value is considered.



Appendix

— Goal of Backpropagation: To optimize the weight

Appendix

— Forward propagation:

Total net input for **h1** (net_{h1}) = w1 * i1 + w2 * i2 + b * 1

net_{h1} = 0.15 * 0.05 + 0.2 * 0.1 + 0.35 * 1 = 0.3775

Squash using sigmoid function (out_{h1}) = $\frac{1}{1+e^{-net_{h1}}} = \frac{1}{1+e^{-0.375}} = 0.5932$

Similarly for **h2**: out_{h2} = 0.5968

Similarly for output **o1**:

net_{o1} = w5 * out_{h1} + w6 * out_{h2} + b2 * 1 = 0.4 * 0.5932 + 0.45 * 0.5968 + 0.6 * 1 = 1.1059

out_{o1} = 0.7513

Appendix

$out_{o2} = 0.7729$

Total Error (E_{total}) = $\sum \frac{1}{2} \ (target \ - output)^2$

$E_{o1} = \frac{1}{2} \ (target_{o1} \ - \ output_{o1})^2 = 0.2748$

$E_{o2} = 0.0235$

$E_{total} = E_{o1} + E_{o2} = 0.2983$

Appendix

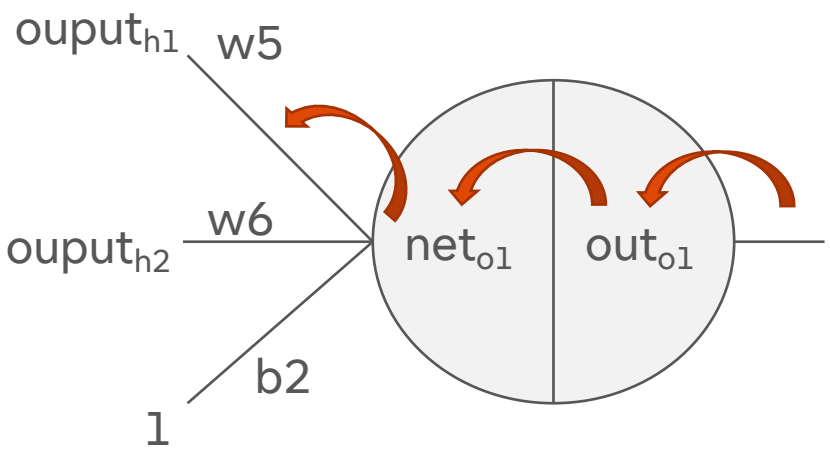
— Backward Propagation:

Output layer:

Consider w5 and how much change in w5 affects the total error. ($\frac{\partial E_{total}}{\partial w5}$)

Applying the chain rule:

$$\frac{\partial E_{total}}{\partial w5} = \frac{\partial E_{total}}{\partial out_{o1}} * \frac{\partial out_{o1}}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial w5}$$



Appendix

$$E_{total} = \frac{1}{2}\ (target_{o1} - output_{o1})^2 + \frac{1}{2}\ (target_{o2} - output_{o2})^2$$

$$\frac{\partial E_{total}}{\partial out_{o1}} = 2 * \frac{1}{2}\ (target_{o1} - output_{o1})^{2-1} * (-1) + 0 = 0.7413$$

$$out_{o1} = \frac{1}{1+e^{-net_{o1}}}$$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1} * (1 - out_{o1}) = 0.1868$$

$$net_{o1} = w5 * out_{h1} + w6 * out_{h2} + b2 * 1$$

$$\frac{\partial net_{o1}}{\partial w5} = 1 * out_{h1} * w5^{(1-1)} + 0 + 0 = 0.5932$$

Appendix

$$\frac{\partial E_{total}}{\partial w5} = \frac{\partial E_{total}}{\partial out_{o1}} * \frac{\partial out_{o1}}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial w5}$$

$$\frac{\partial E_{total}}{\partial w5} = 0.7413 * 0.1868 * 0.5932 = 0.0821$$

$$w5 = w5 - \eta * \frac{\partial E_{total}}{\partial w5} = 0.4 - 0.5 * 0.0821 = 0.3581$$

Appendix

$$\frac{\partial E_{total}}{\partial w1} = \frac{\partial E_{total}}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w1}$$

$$\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$

$$w1 = w1 - \eta * \frac{\partial E_{total}}{\partial w1} = 0.1497$$