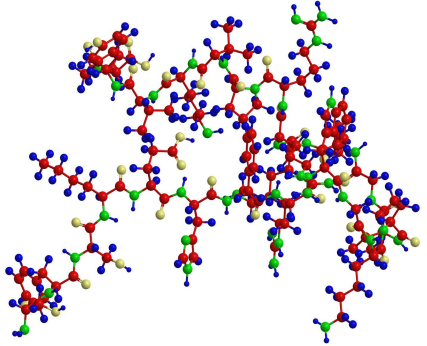


Protein function prediction using Recurrent Neural Network

Nucleic Acids

- DNA(Deoxyribonucleic acid)
- RNA(Ribonucleic acid)

Nucleic acids are made of nucleotides. Nucleotides hold genetic information and variety of other informations.

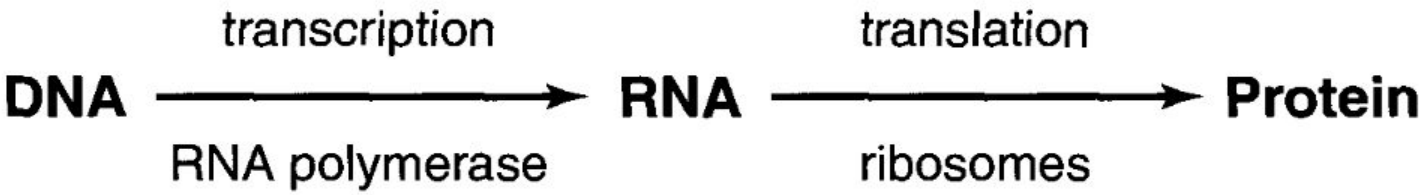
Nucleotides of DNA

1. Cytosine(C)
2. Thymine (T)
3. Adenine (A)
4. Guanine (G)

Nucleotides of RNA

1. Cytosine(C)
2. Uracil(U)
3. Adenine (A)
4. Guanine (G)

- NUCLEIC ACIDS are made up of NUCLEOTIDES.
- PROTEINS are made up of AMINO ACIDS.

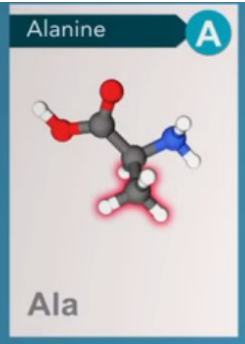


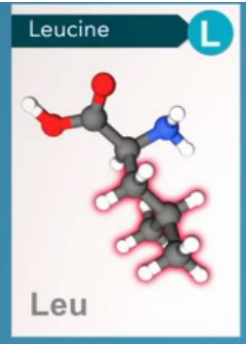
Central Dogma of molecular biology

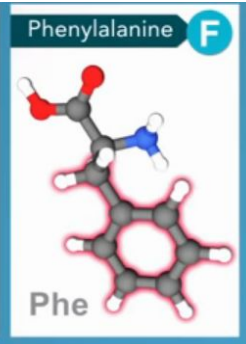
THE BUILDING BLOCKS

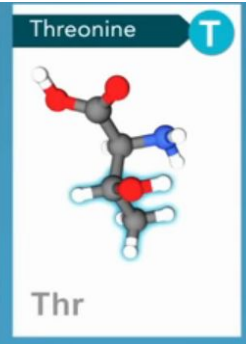
RCSB PDB-101

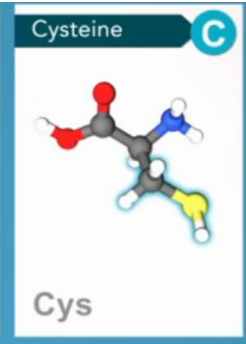
STANDARD AMINO ACIDS

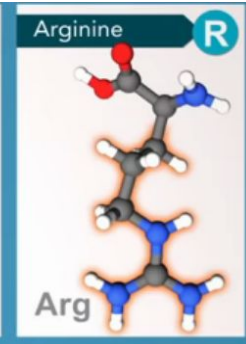


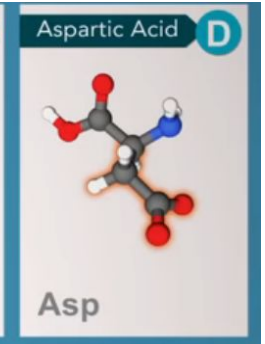


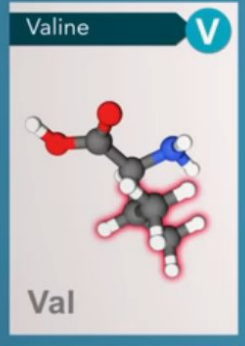


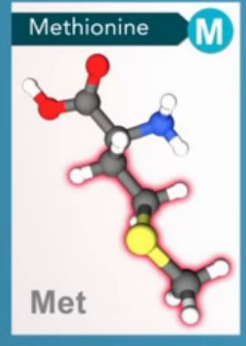


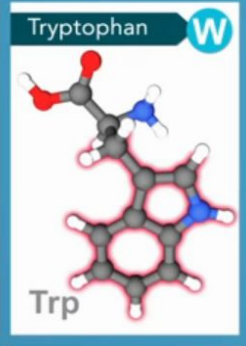


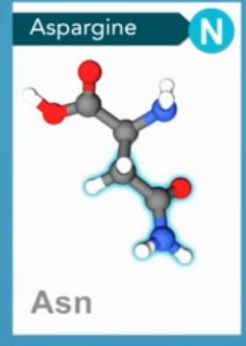


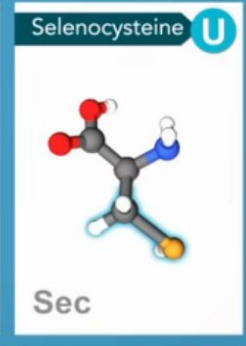


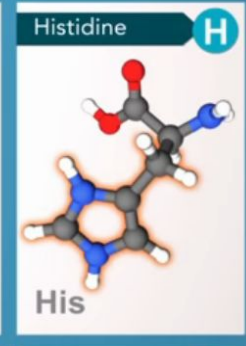


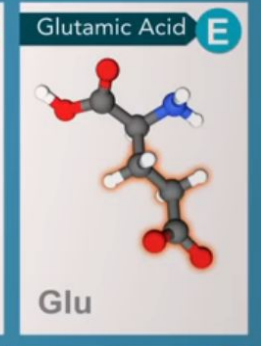


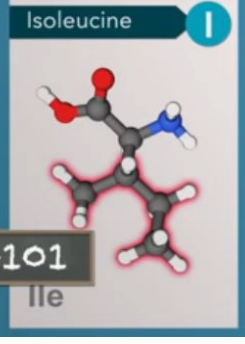


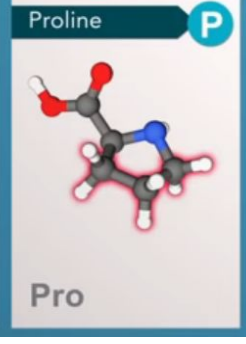


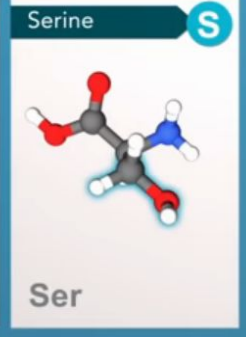


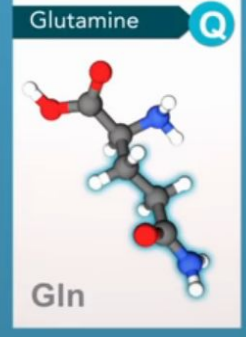


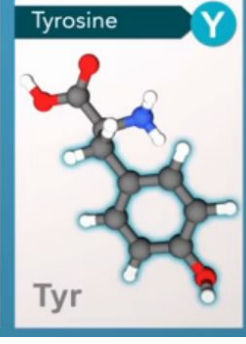


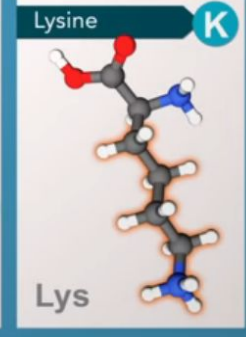


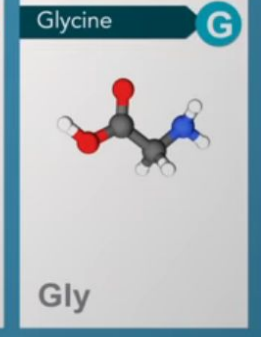












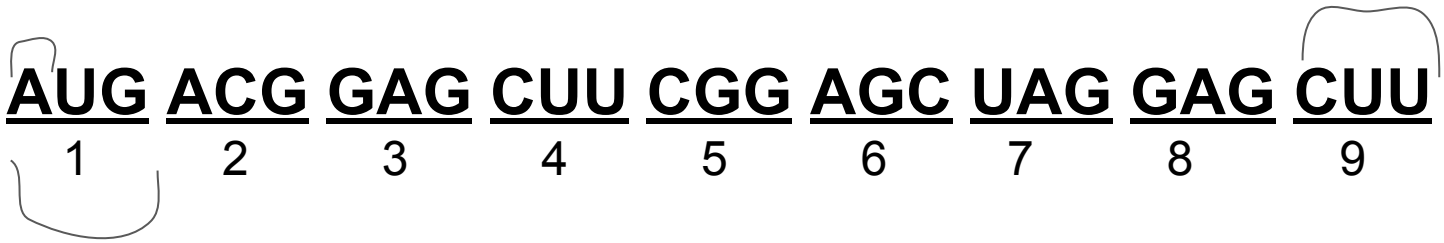
RCSB PDB-101

AUGACGGAGCUUCGGAGCUAGGAGCUU

RNA sequence

codons

Nucleotide



Amino acid

Protein

Codons

- A codon is a sequence of three DNA or RNA nucleotides that corresponds with a specific amino acid or stop signal during protein synthesis.
- Each codon corresponds to a single amino acid (or stop signal).
- Codon is a triplet. Out of the 64 codons, 61 codons code for 20 amino acids and 3 codons (UAA, UGA and UAG) do not code for any amino acids. Thus, they function as terminating codons.

In general, the following methods are used for protein function prediction.

- Basic Local Alignment Search Tool (BLAST)
- Network based methods
- Information based methods

Problem statement

The computation method that could accurately and quickly predict protein function from its sequence.

Recurrent Neural Network

- Recently gaining popularity and successes for natural languages processing
- Use protein sequence without searching any database
- Capture complex patterns in biological sequences in order to predict protein functions, potentially beyond the capability of current methods.

Dataset

For training - UniProt Knowledgebase (UniProtKB)

- **Protein sequence**- MAFSAEDVLKEYDRRRRMEALLLSLYYPNDRKLLDYKEW
- **GO:ID** - GO:0000016

For testing - CAFA3(The Critical Assessment of Function Annotation)

- **Protein sequence**-
  MASNTVSAQGGSNRPVRDFSNIQDVAQFLLFDPIWNEQPGSIVPWKMNREQALAERYPEL

Approach

1. Generating “ProLan” language by extracting protein “word” from protein sequence.

2. Converting protein “word” into fixed-size vector.

3. Converting “GOLan” language into textual data.

4. Predicting GOLan using RNN model.

k-mers

k-mers are subsequences of length k contained within a biological sequence.

***k*-mers for GTAGAGCTGT**

k	*k*-mers
1	G, T, A, G, A, G, C, T, G, T
2	GT, TA, AG, GA, AG, GC, CT, TG, GT
3	GTA, TAG, AGA, GAG, AGC, GCT, CTG, TGT
4	GTAG, TAGA, AGAG, GAGC, AGCT, GCTG, CTGT
5	GTAGA, TAGAG, AGAGC, GAGCT, AGCTG, GCTGT
6	GTAGAG, TAGAGC, AGAGCT, GAGCTG, AGCTGT
7	GTAGAGC, TAGAGCT, AGAGCTG, GAGCTGT
8	GTAGAGCT, TAGAGCTG, AGAGCTGT
9	GTAGAGCTG, TAGAGCTGT
10	GTAGAGCTGT

Algorithm used to divide protein sequence into a set of k-mers or protein “word”

Scan the whole training dataset - UniProtKB knowledge database to get a protein “word” database, which includes all k-mers whose frequency f^K is larger than 1,000, where $k \in [3, 5]$.

3	GTA, TAG, AGA, GAG, AGC, GCT, CTG, TGT
4	GTAG, TAGA, AGAG, GAGC, AGCT, GCTG, CTGT
5	GTAGA, TAGAG, AGAGC, GAGCT, AGCTG, GCTGT

Why we have used the range of $k \in [3, 5]$?

Gene Ontology (GO):

- Representation of gene and gene product attributes across all the species.
- Focuses on the function of genes and gene products.
- Traditional methods directly predict GO terms for a protein sequence.

Gene Ontology (GO) contd:

- GO term:
  - Represented by GO term ID using a seven digit number
  - has a term name and a namespace indicating the domain to which it belongs.

Gene Ontology (GO) contd:

Three biological domains:

1. Biological Processes - Operations or sets of molecular events with a defined beginning and end, pertinent to the functioning of integrated living units.
2. Cellular component - The parts of a cell or its extracellular environment
3. Molecular function - The elemental activities of a gene product at the molecular level.

Gene Ontology (GO) contd:

- Example term:

  Id : GO:0000016

  Name: lactase_activity

  Namespace: molecular_function

- The GO ontology is freely available from the GO website.

26 Base Alphabet ID

- We assign 26-base alphabet number to each GO term as new ID(Alphabet ID).
- Rules of Conversion:

  We represent each alphabet from A to Y as numbers from 1 to 25 respectively and Z as 0. So for example $(ayv)_{26}$

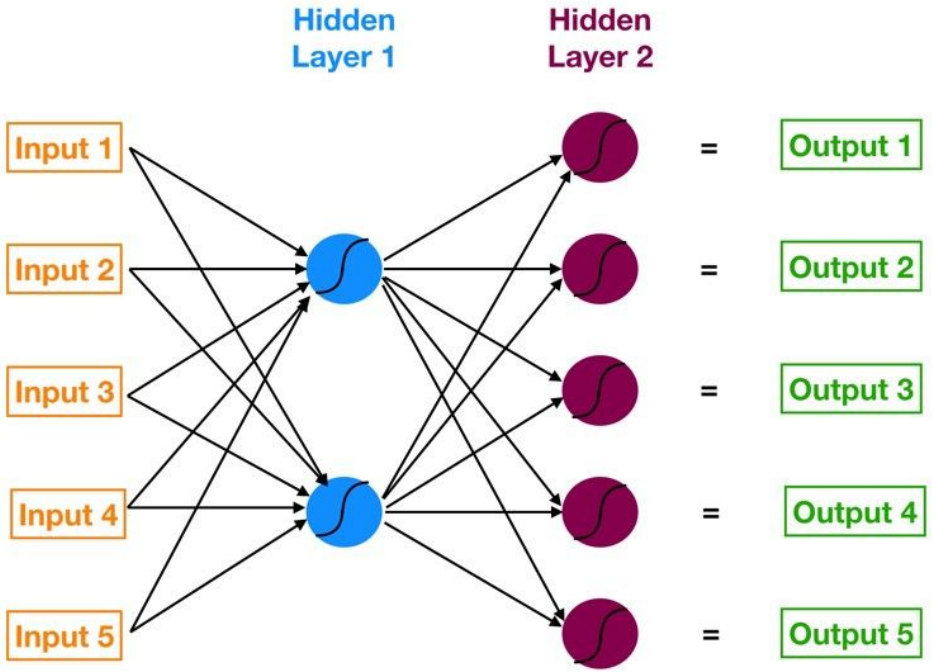
  Can be converted to decimal form as: $(26^2 \times 1) + (26^1 \times 25) + (26^0 \times 22) = 1348$

We have used word2vec model to encode the k-mers to vector form to give input to the RNN.

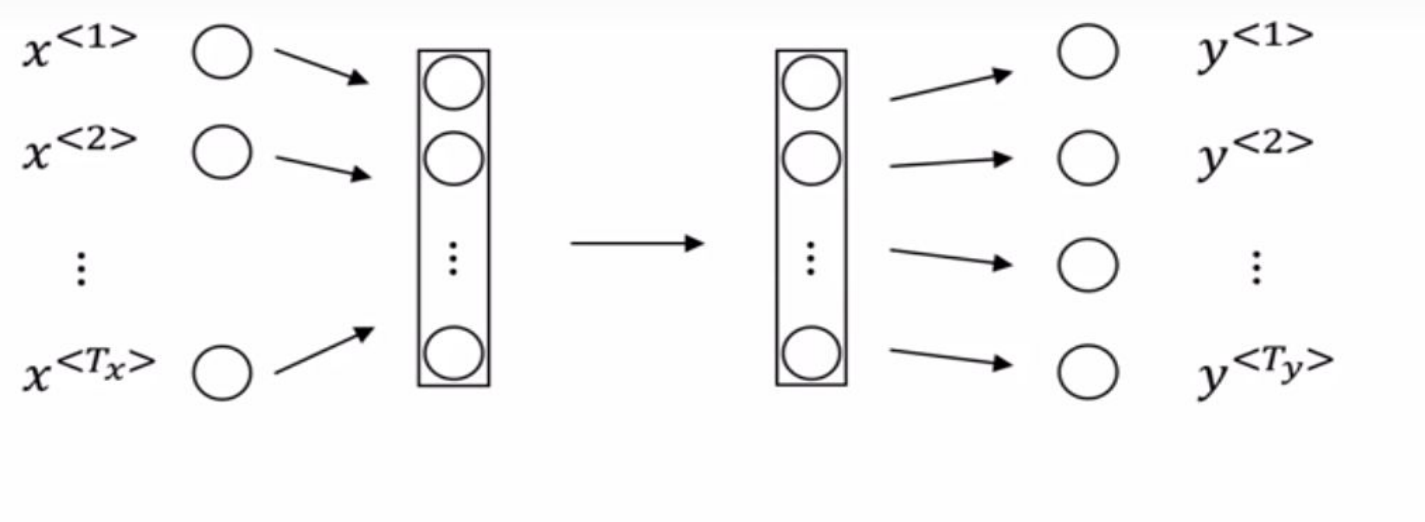
- Capable of capturing:
  - context of a word in a document
  - semantic and syntactic similarity
  - relation with other words, etc

Neural Networks :

Inputs are I.I.D



Why not standard neural network?

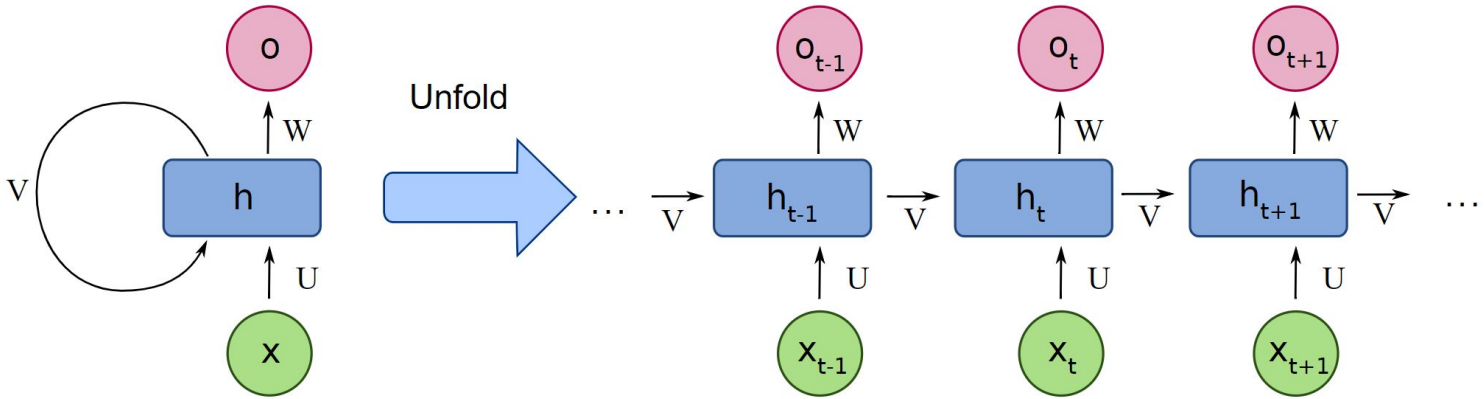


Problems

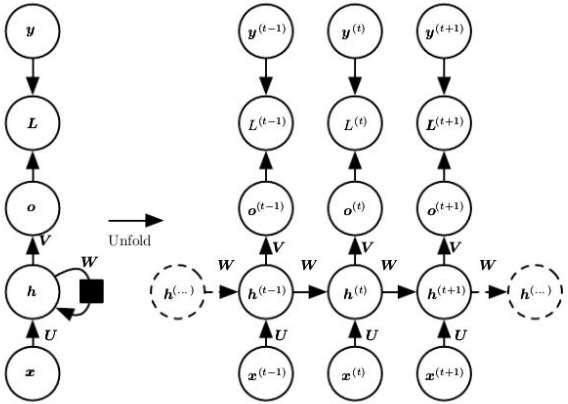
- Inputs and Outputs can be of different length in different examples.
- Does not share features learned across different position of text.

Solution:

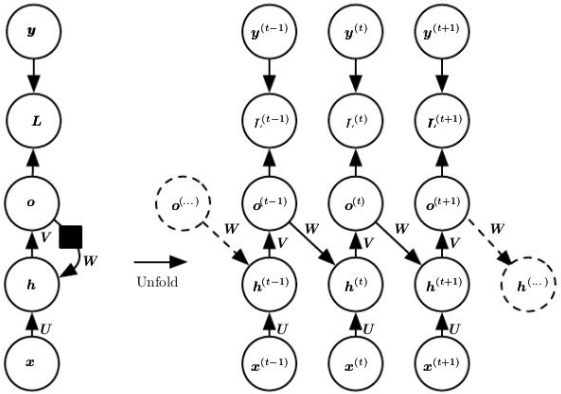
- Output from previous steps are fed as input to the current step.
- Has Hidden State which remembers information about a sequence.
- Have memory
- $h_t = f(h_{t-1}, x_t)$



Recurrent neural network:(Example)



(a)



(b)

It is very useful when we have sequential data.

Ex :

- Nucleotide base pairs in a strand of DNA
- Sequence of characters in an English sentence
- Parts of speech of successive words

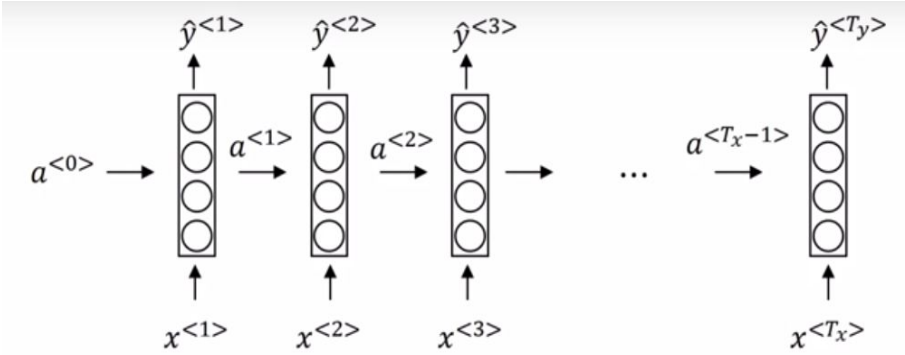
Forward Propagation :

Initialize: $a^{<0>} = 0$

$a^{<t>} = g(W*a^{<t-1>} + U*x_t + b)$

$y\_p^{<t>} = g(c+V*a^{<t>})$

- $W_{(h*h)}$,$U_{(h*n)}$,$V_{(m*h)}$ are parameters and b,c are bias.
- n and m are length of input and output vector respectively i.e *numbers of words in vocabulary.*
- h is number of neuron in hidden layer.
- $a^{<t>}$: (h*1)
- Activation function : tanh, Relu



Cost Function:

$L(y^{<t>}, y\_p^{<t>}) = -(y^{<t>} * \log(y\_p^{<t>}) + (1-y^{<t>}) * \log(1-y\_p^{<t>}))$

Here $T_x = T_y = T$, so :

$C(y, y\_p) = \Sigma^{T}_{t=0} L(y^{<t>}, y\_p^{<t>})$

Backpropagation through time:

Gradient :

$$\frac{\partial L}{\partial U} = \frac{\partial L}{\partial y\_p^{<t>}} * \frac{\partial y\_p^{<t>}}{\partial a^{<t>}} * \frac{\partial a^{<t>}}{\partial U}$$

$$\frac{\partial L}{\partial V} = \frac{\partial L}{\partial y\_p^{<t>}} * \frac{\partial y\_p^{<t>}}{\partial V}$$

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial y\_p^{<t>}} * \frac{\partial y\_p^{<t>}}{\partial a^{<t>}} * \frac{\partial a^{<t>}}{\partial a^{<t-1>}} * \ldots * \frac{\partial a^{1}}{\partial W}$$

Recurrent neural network offers a lot of flexibility

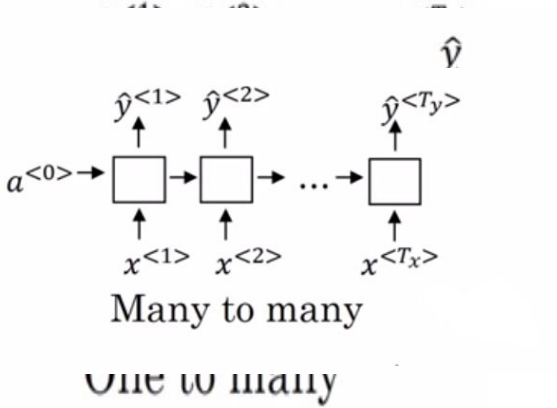


Image Captioning (image -> Sequence of words)

Sentiment classification (sequence -> sentiment)

Machine translation (seq of words -> seq of words)

Conclusion

- In this project, we propose a novel language model ProLanGO for the protein function prediction problem.
- We first convert protein sequences into a language space “ProGO” based on the frequency of k-mers.
- Then, Gene Ontology terms into a language space “LanGO”.
- In addition, we convert the protein function prediction problem to a language translation problem.

References

ProLanGO: Protein Function Prediction Using Neural Machine Translation Based on a Recurrent Neural Network

https://www.ncbi.nlm.nih.gov/pubmed/29039790

https://arxiv.org/pdf/1701.08318.pdf

https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa