신경망 기반 자동번역 기술

김강일

Konkuk University

Computational Intelligence Lab.

http://ci.konkuk.ac.kr
kikim01@kunkuk.ac.kr

Index

Issues in AI and Deep Learning

Issues in AI and Deep Learning

1. What is the distinguished property of deep learning?

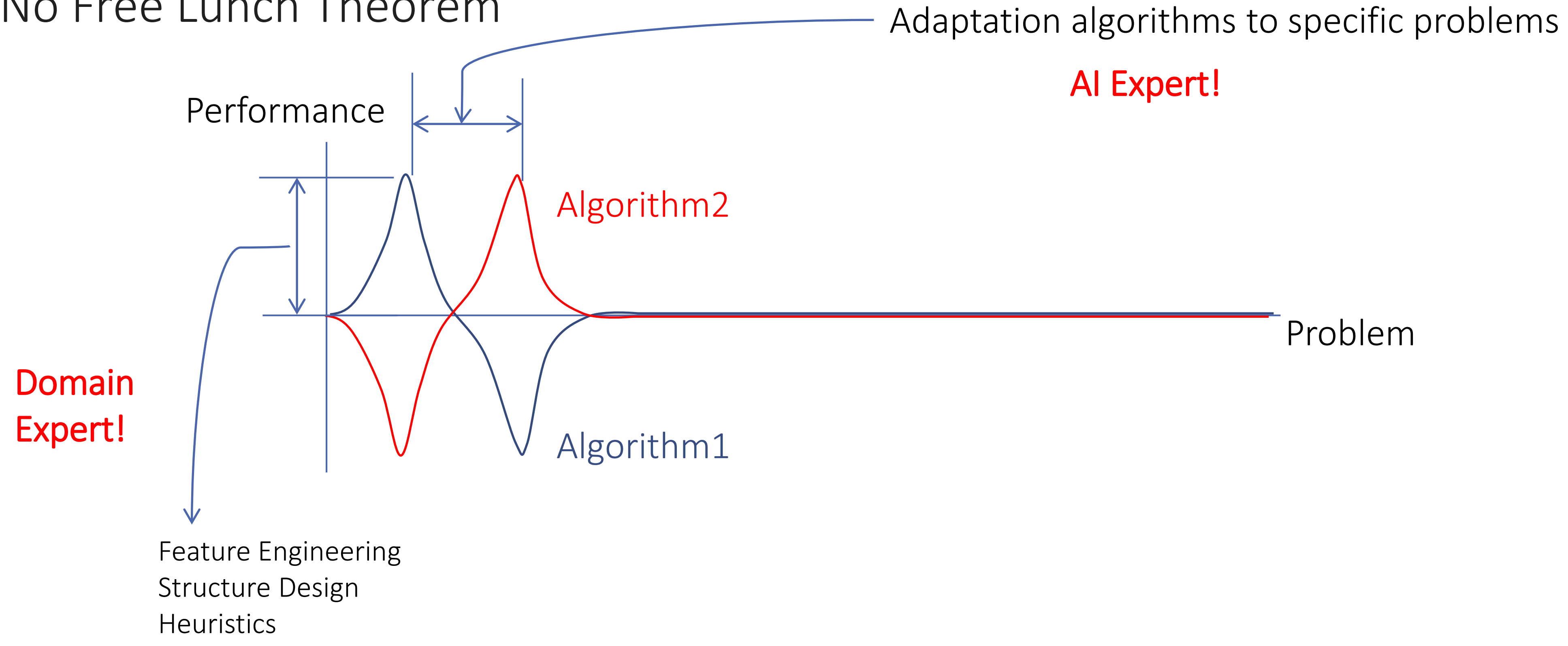
2. What is the range of problems solved by deep learning?

3. Why deep learning can abstract features?

4. Why deep learning can extract features?

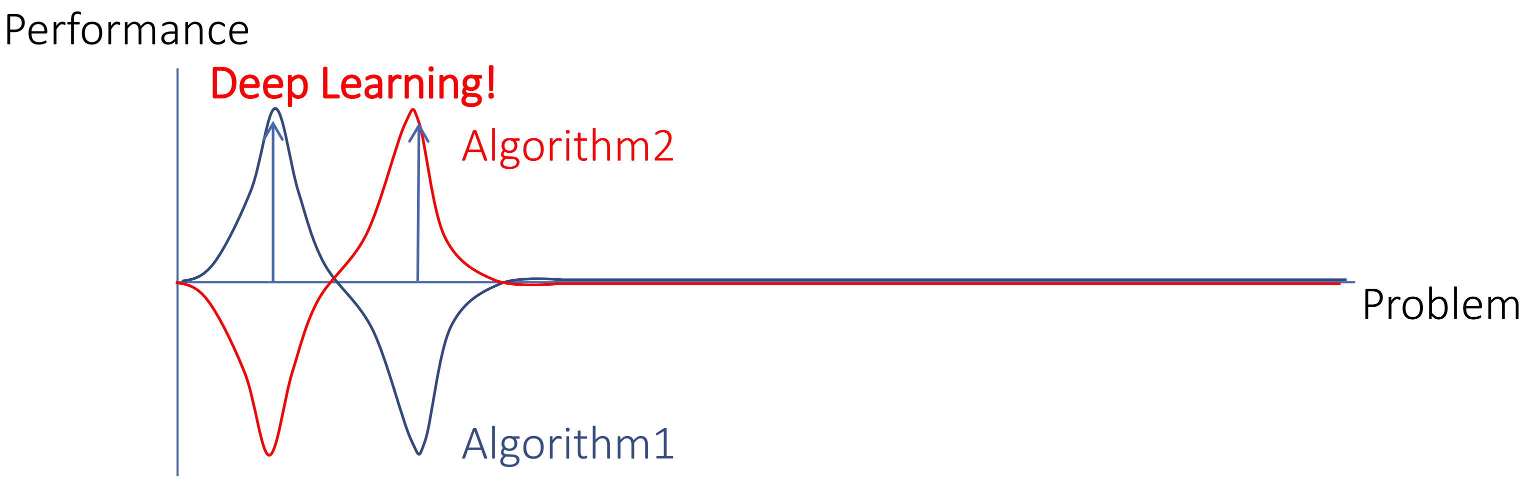
Issues in AI and Deep Learning

No Free Lunch Theorem



Issues in AI and Deep Learning

No Free Lunch Theorem



Benefit: (almost) Automated AI System Building

Very good for industrialization

Issues in AI and Deep Learning

1. What is the distinguished property of deep learning?

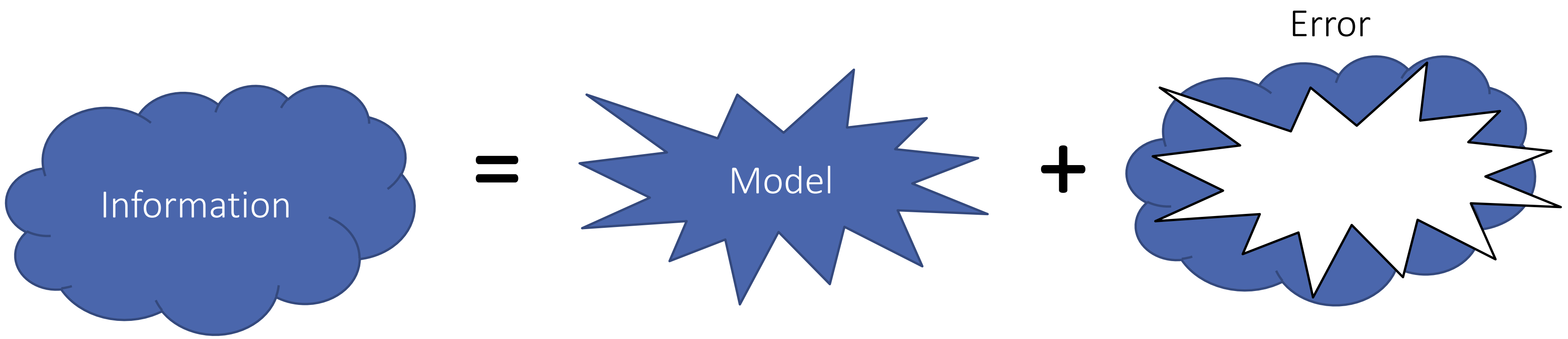
2. What is the range of problems solved by deep learning?

3. Why deep learning can abstract features?

4. Why deep learning can extract features?

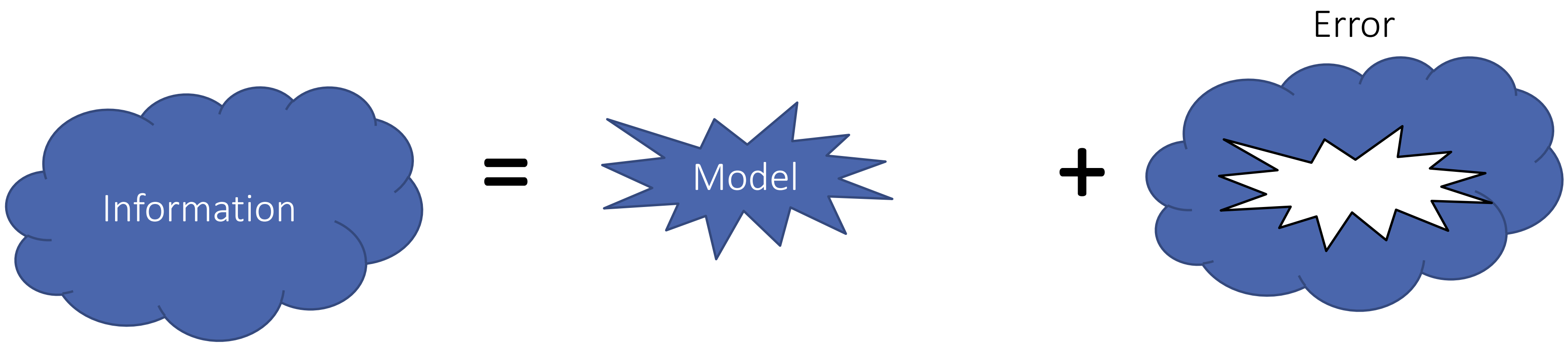
Issues in AI and Deep Learning

To represent information.. (minimum description length..)



Issues in AI and Deep Learning

To represent information.. (minimum description length..)



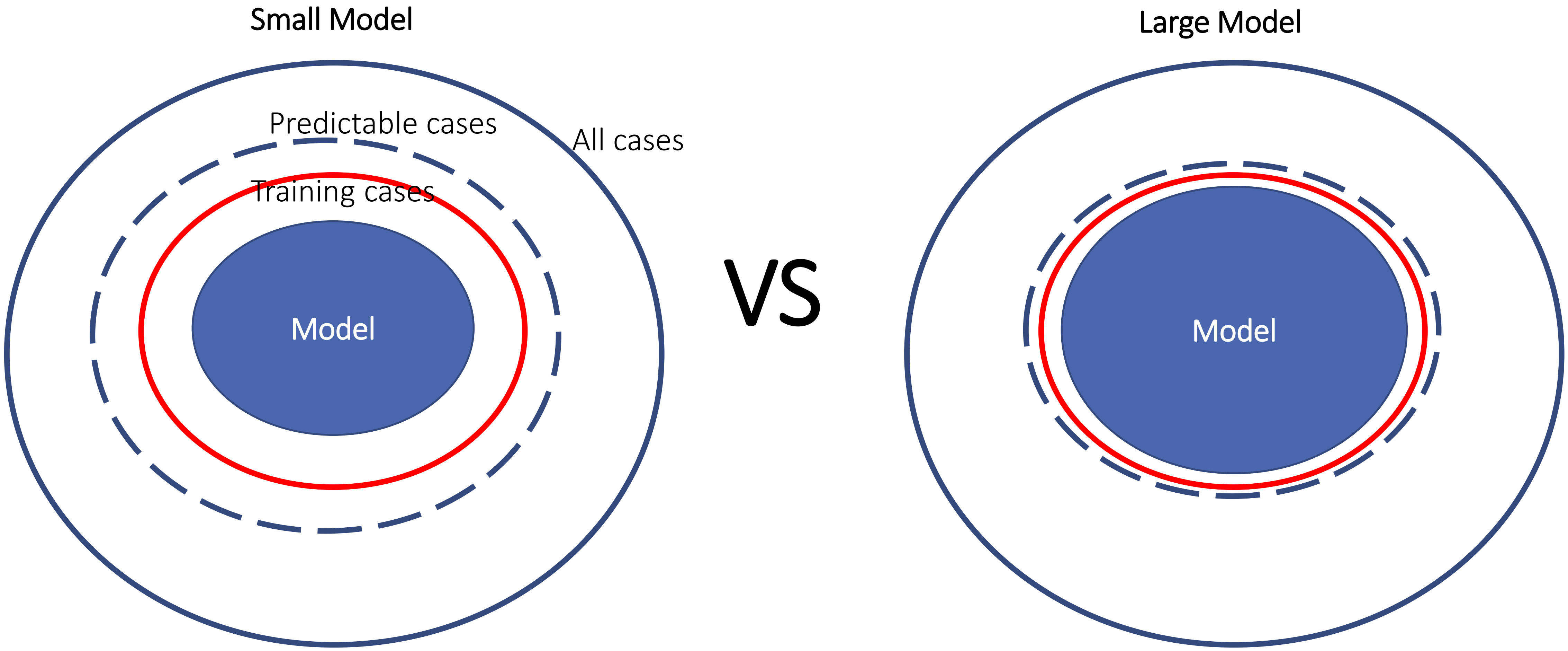
Issues in AI and Deep Learning

Small Model
- Good for representing information as regular patterns
- May restrict representing very complex patterns by implicit model constraints
- Simplified pattern is better for unseen prediction (Belief…)

VS

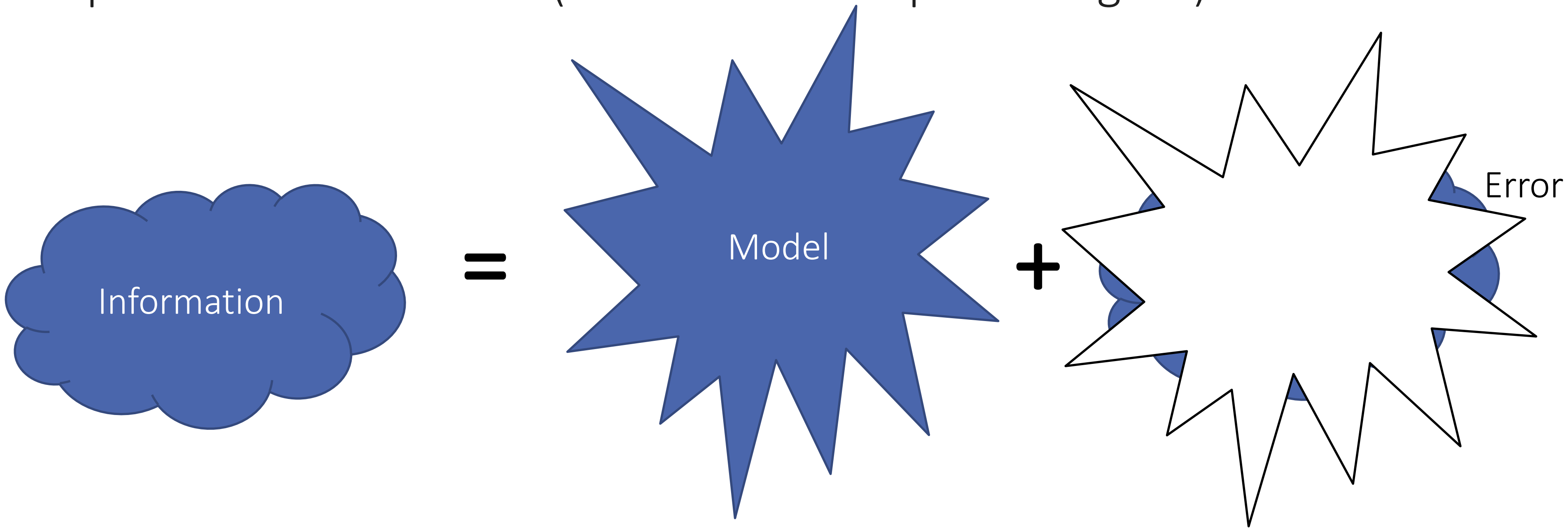
Large Model
- Good for representing all patterns
- Only represent the observed patterns (overfitting)

Issues in AI and Deep Learning



Issues in AI and Deep Learning

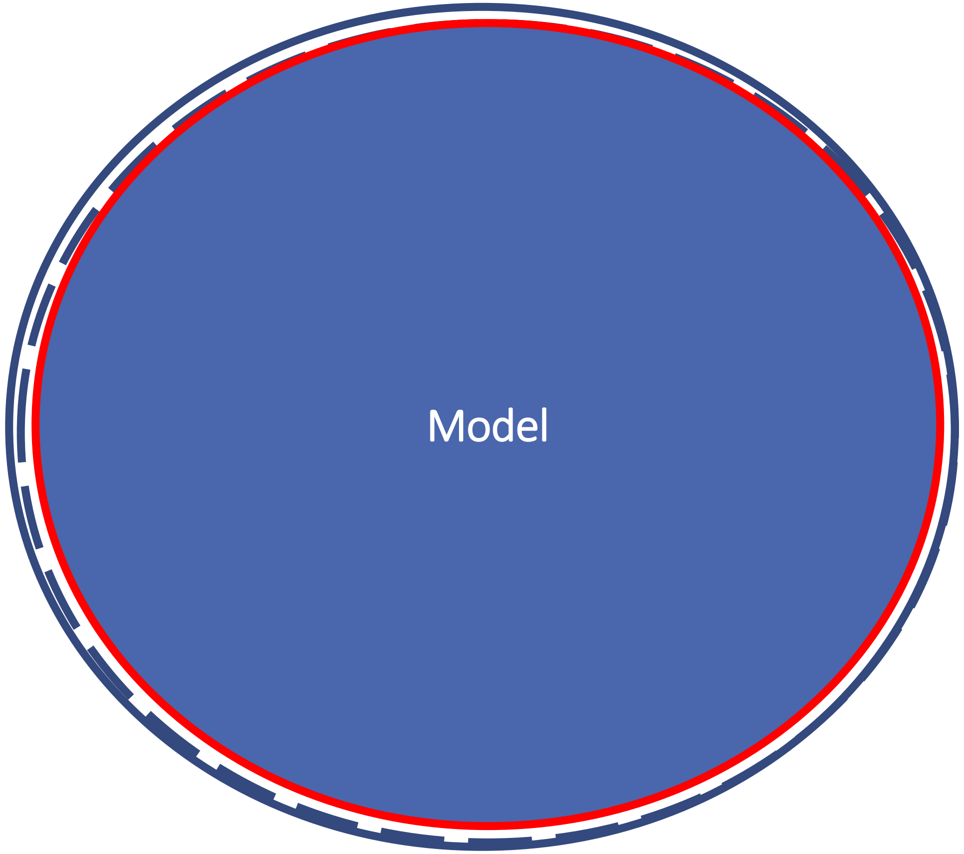
To represent information.. (minimum description length..)



Neural networks are good for representing very accurate and large size models

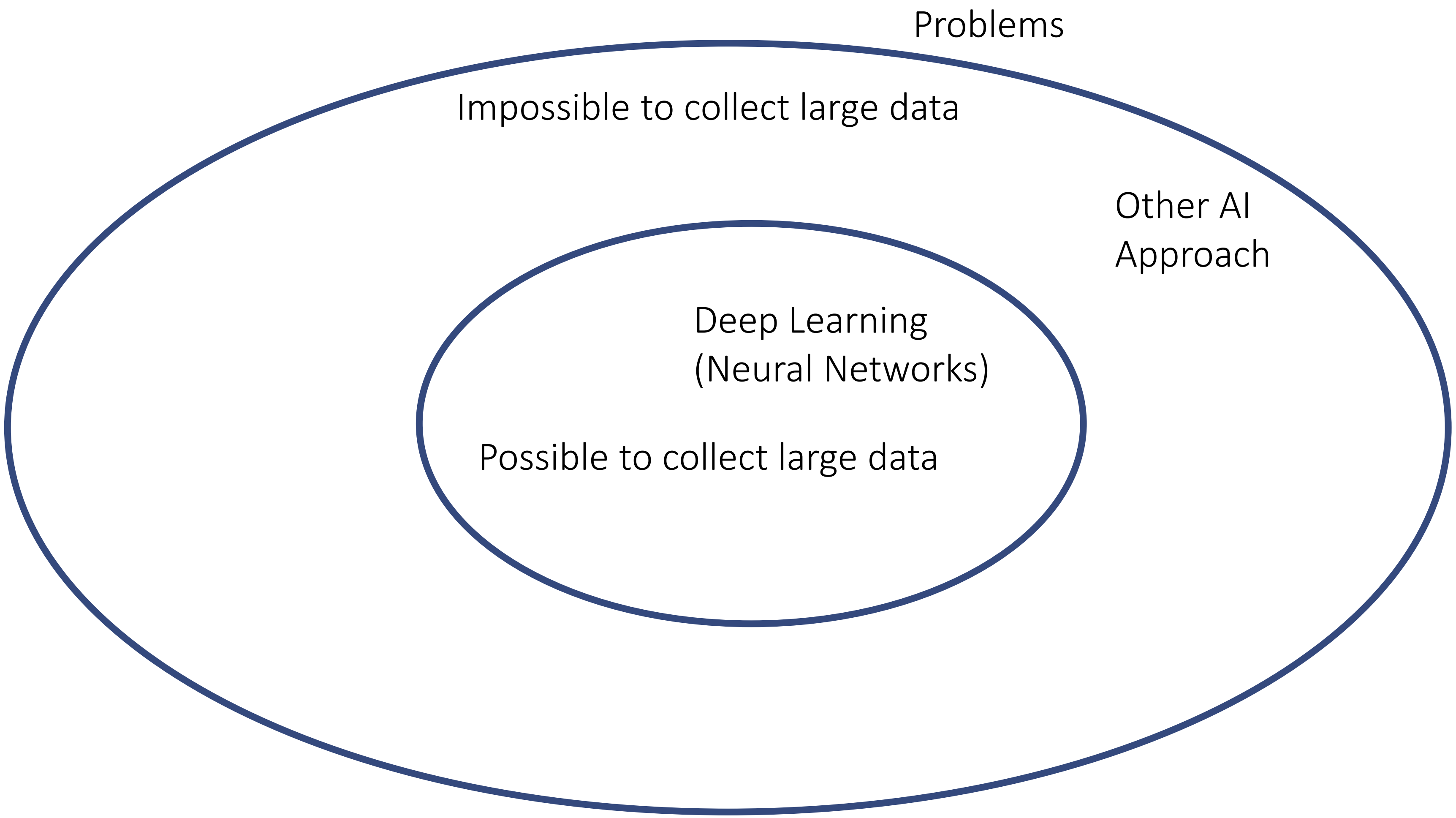
Issues in AI and Deep Learning

Overfitting? -> collect more and more data



Collect Data!!!

Issues in AI and Deep Learning



Issues in AI and Deep Learning

1. What is the distinguished property of deep learning?

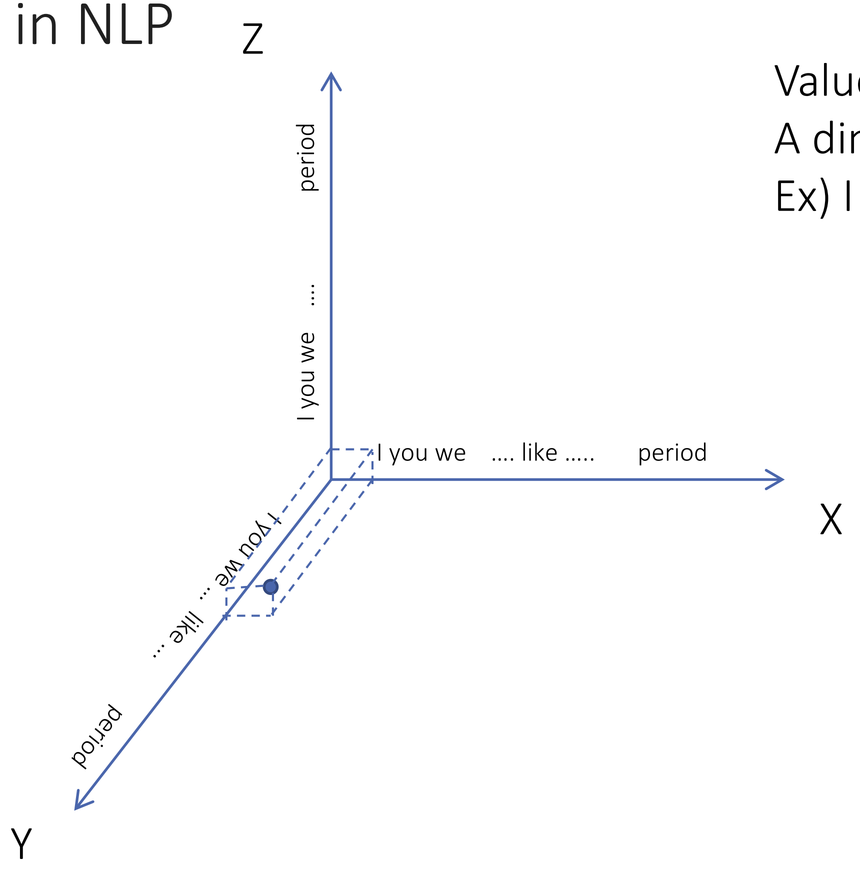
2. What is the range of problems solved by deep learning?

3. Why deep learning can abstract features?

4. Why deep learning can extract features?

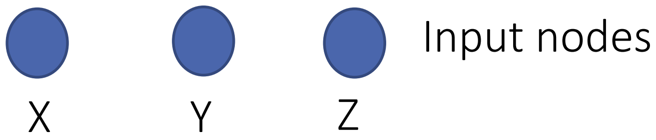
Issues in AI and Deep Learning

Simple example in NLP



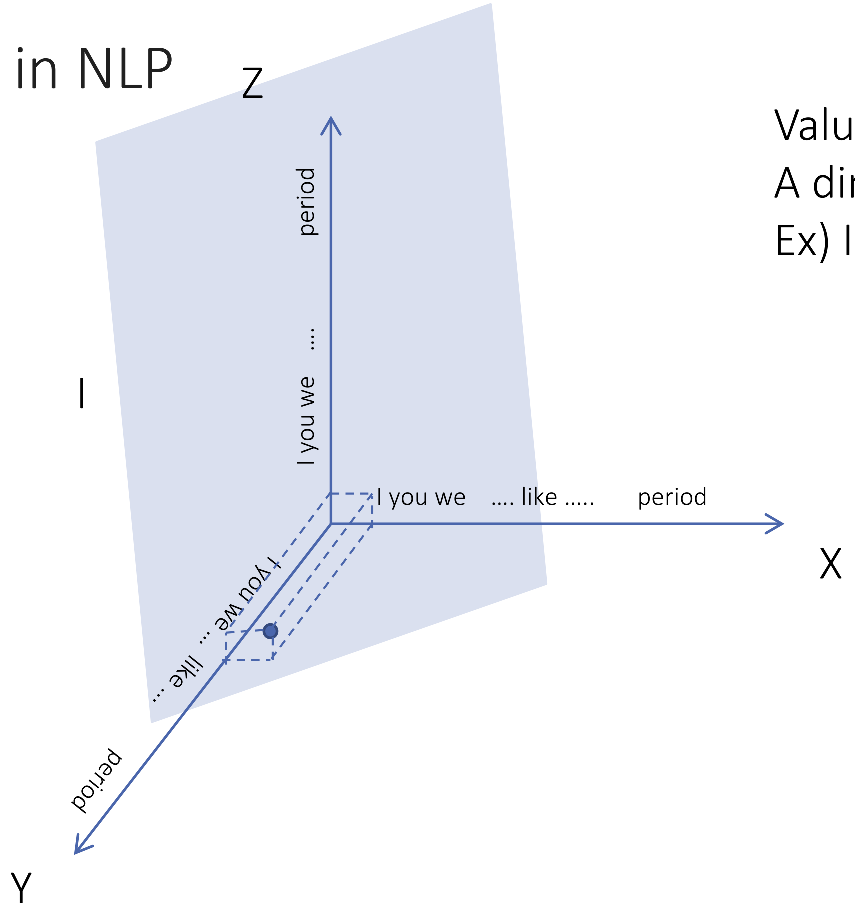
Input Vector Space

Value of each dimension: a word
A dimension: whole vocabulary
Ex) I you we love like on in for .... period



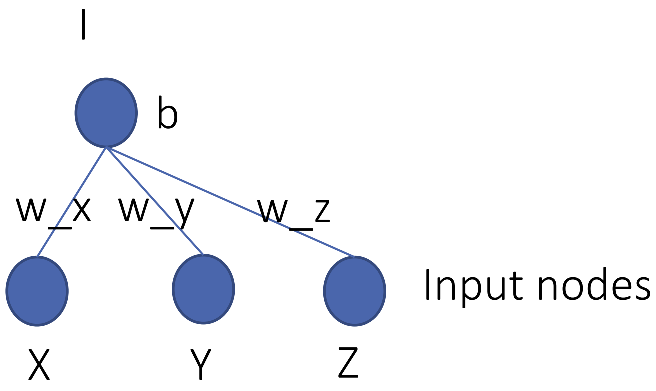
Issues in AI and Deep Learning

Simple example in NLP



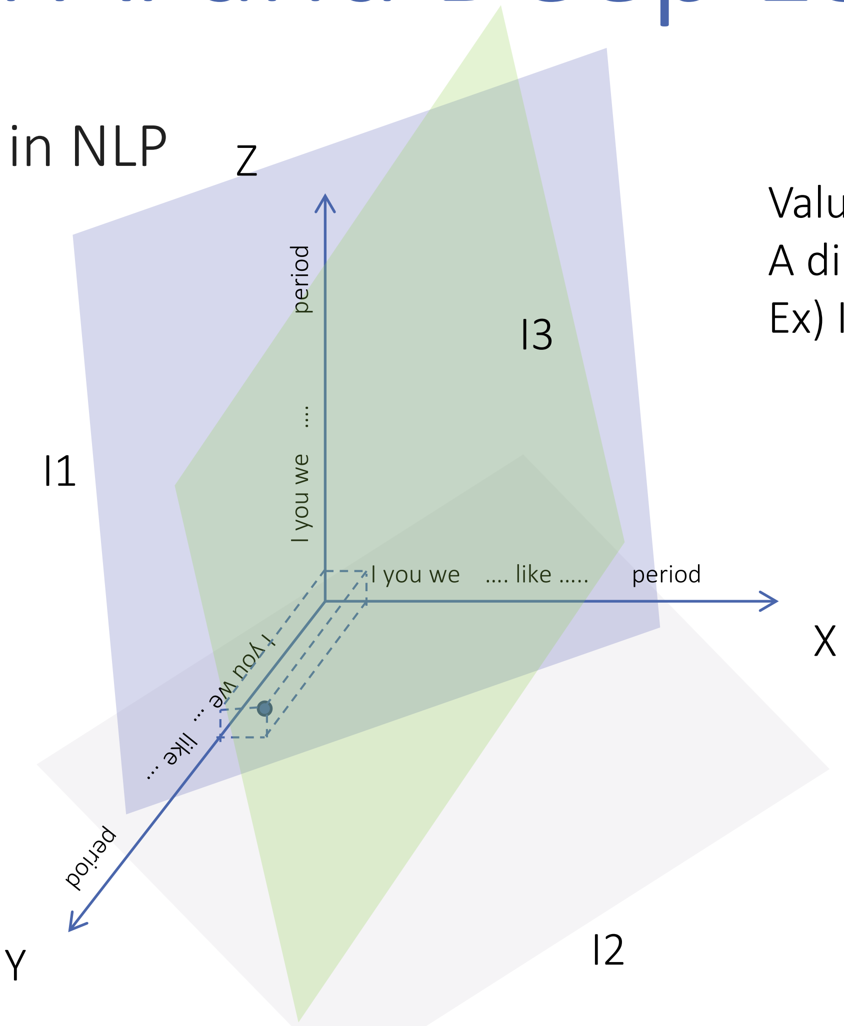
Input Vector Space

Value of each dimension: a word
A dimension: whole vocabulary
Ex) I you we love like on in for …. period



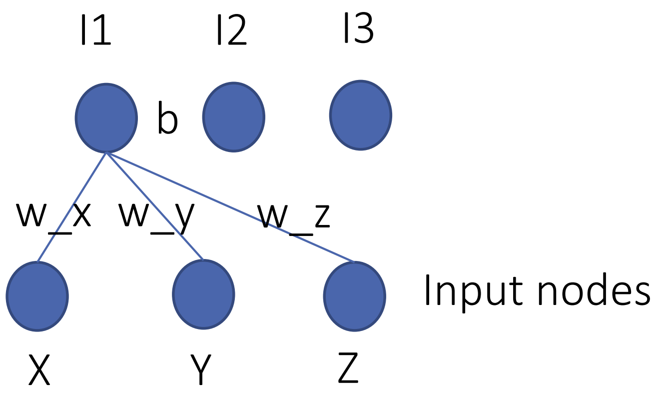
Issues in AI and Deep Learning

Simple example in NLP



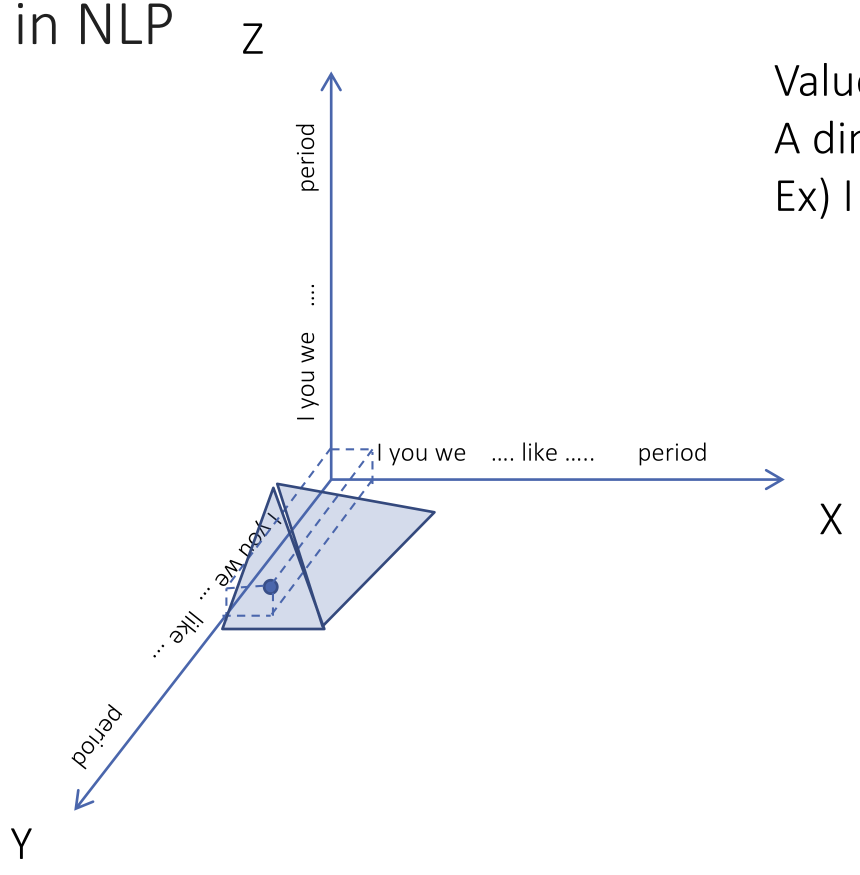
Input Vector Space

Value of each dimension: a word
A dimension: whole vocabulary
Ex) I you we love like on in for …. period



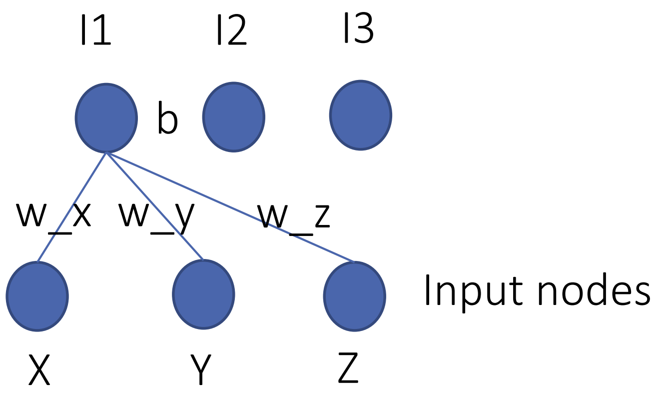
Issues in AI and Deep Learning

Simple example in NLP



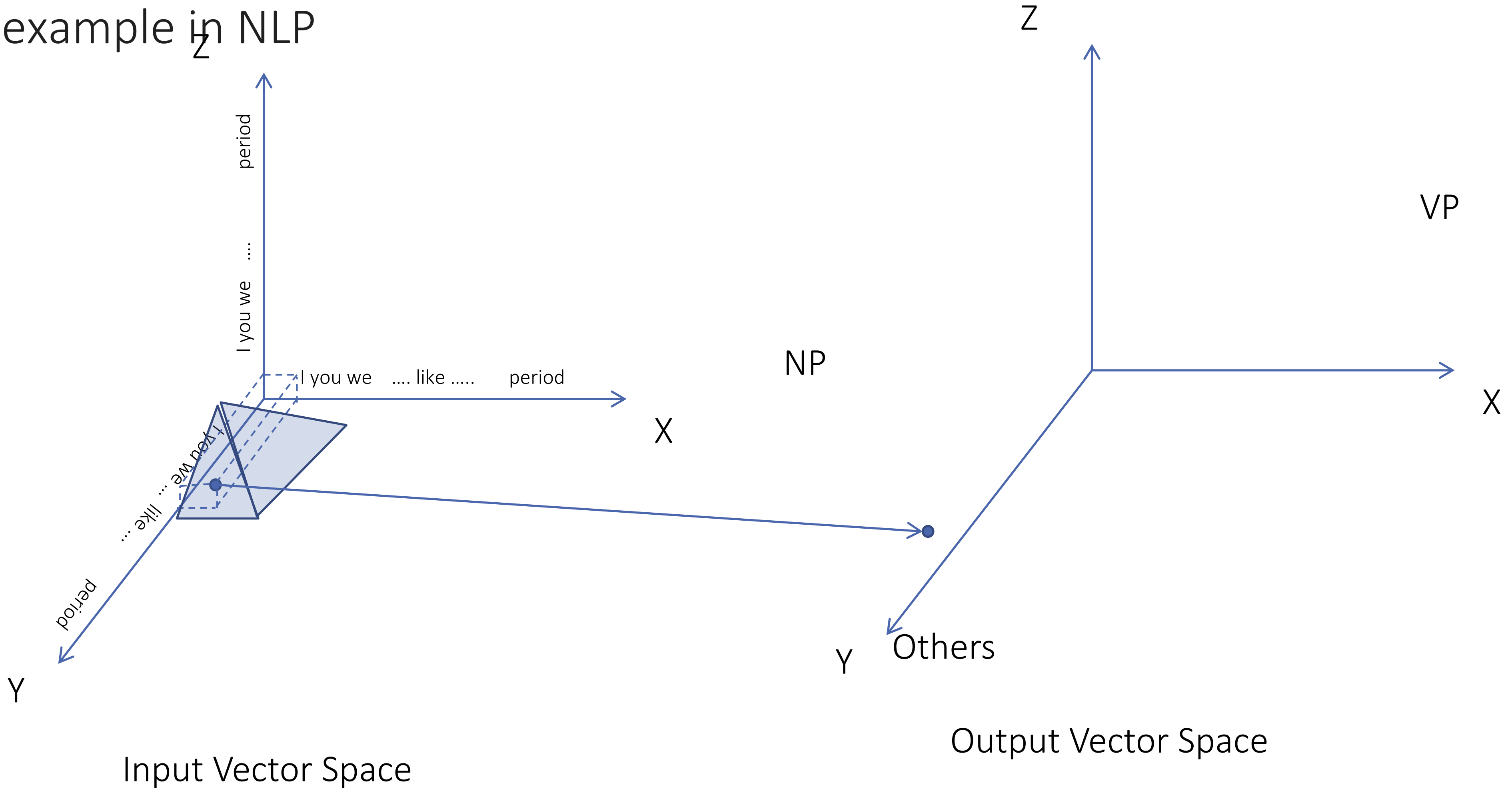
Input Vector Space

Value of each dimension: a word
A dimension: whole vocabulary
Ex) I you we love like on in for …. period



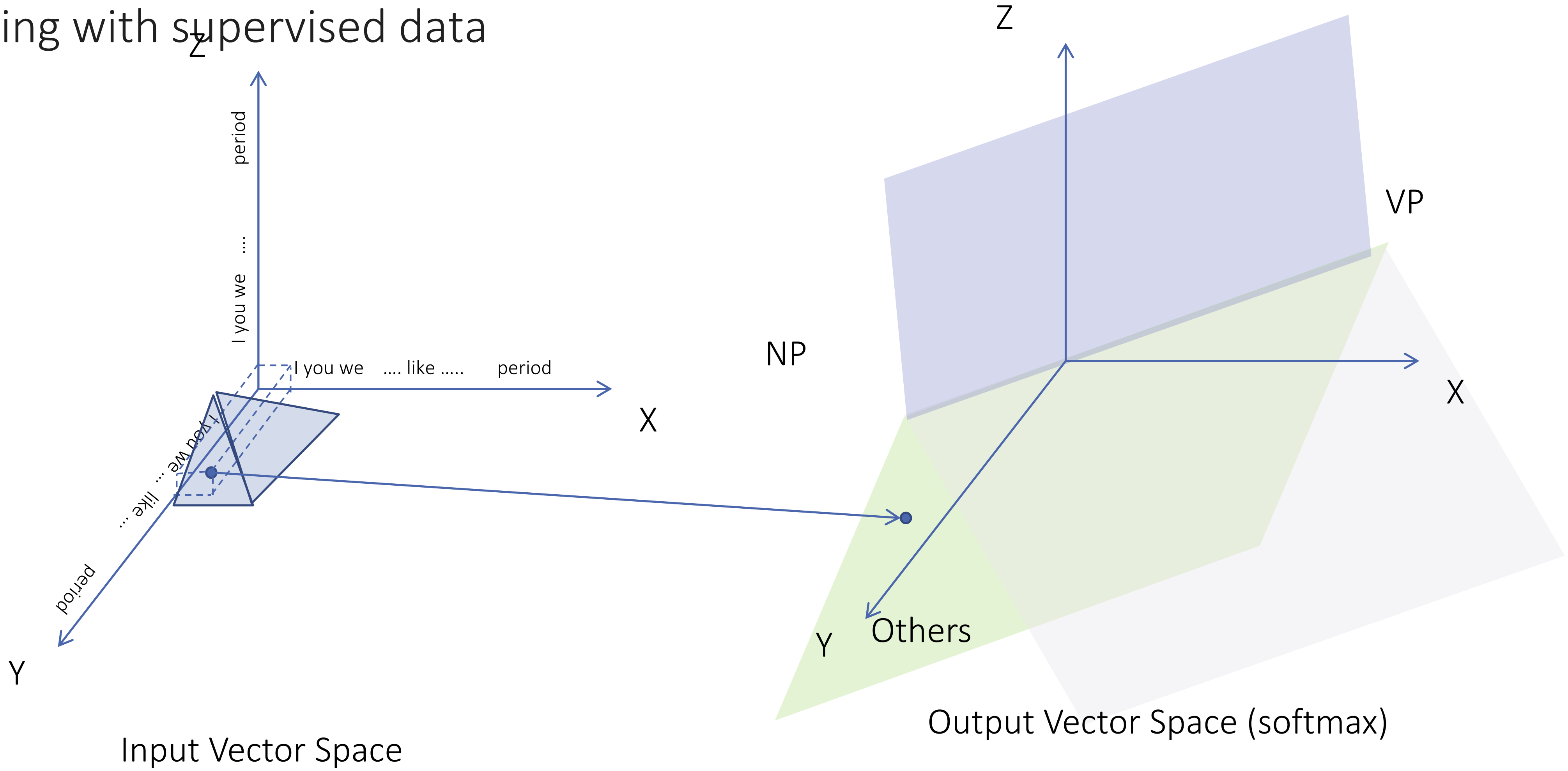
Issues in AI and Deep Learning

Simple example in NLP



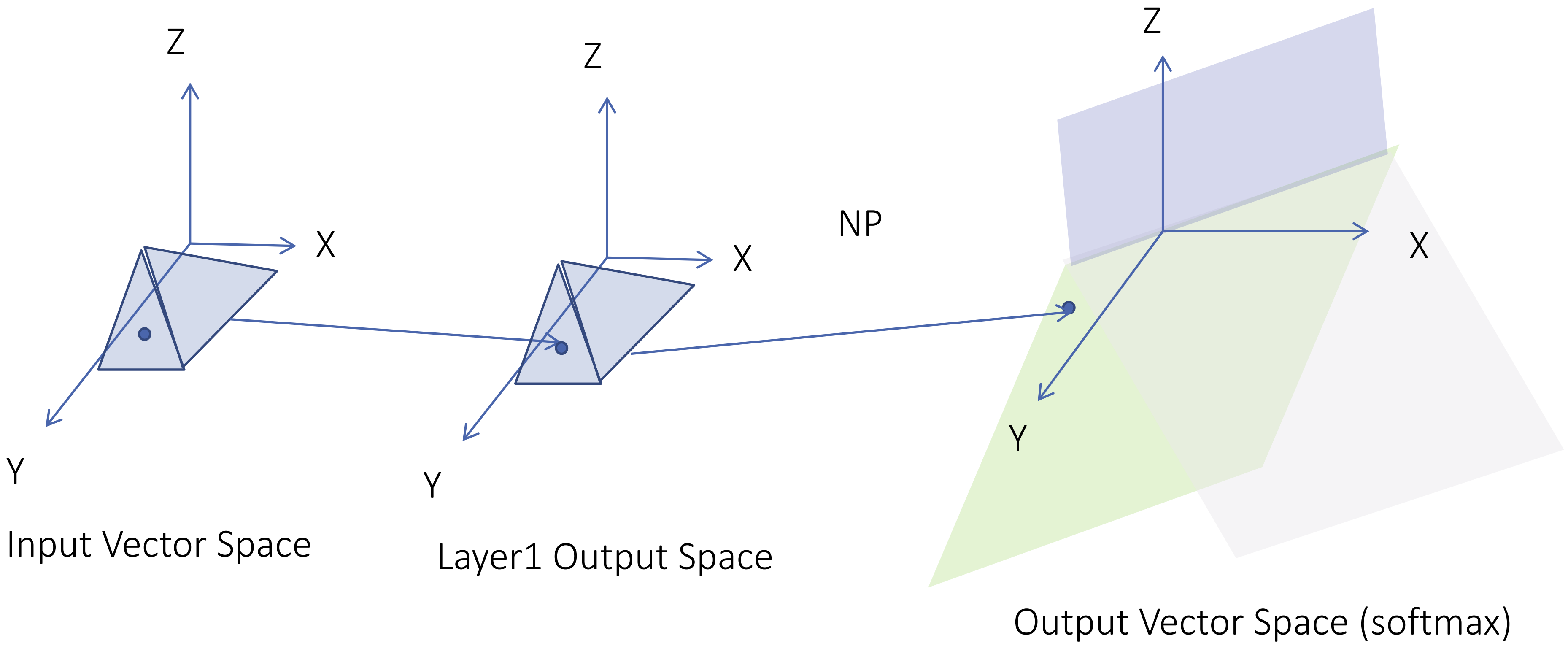
Issues in AI and Deep Learning

In training with supervised data



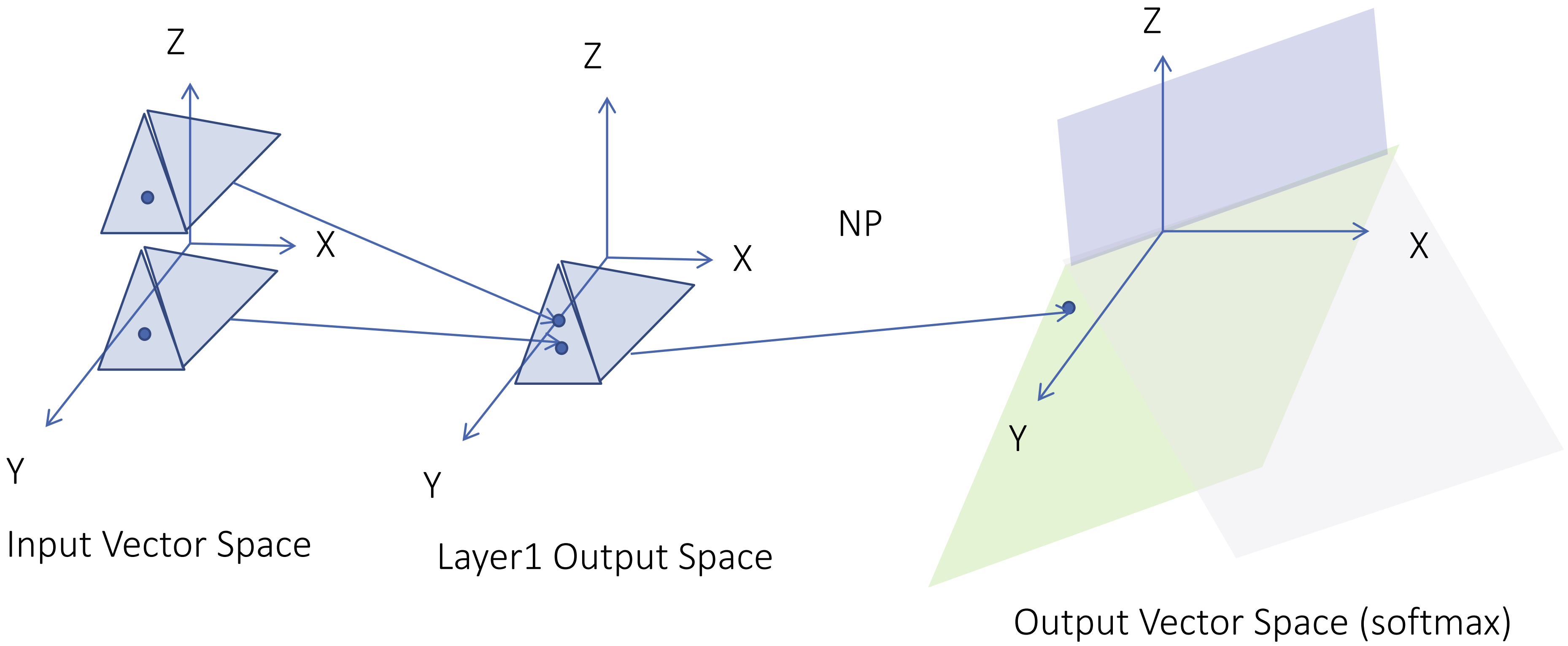
Issues in AI and Deep Learning

In two layers



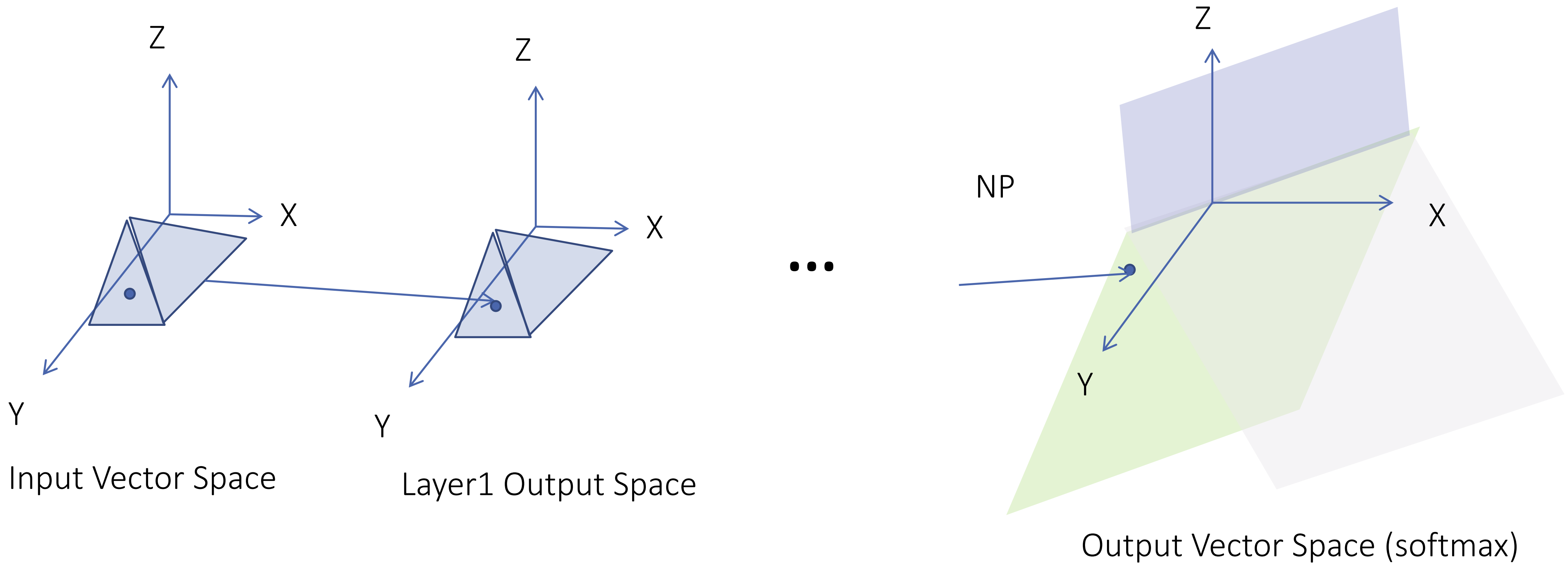
Issues in AI and Deep Learning

Feature abstraction



Issues in AI and Deep Learning

In many layers



Issues in AI and Deep Learning

1. What is the distinguished property of deep learning?

2. What is the range of problems solved by deep learning?

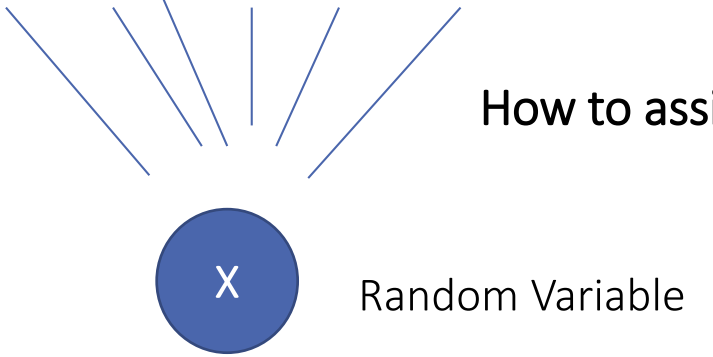
3. Why deep learning can abstract features?

4. Why deep learning can extract features?

Issues in AI and Deep Learning

Compared to a generative probabilistic graphical model?

I want to go to school



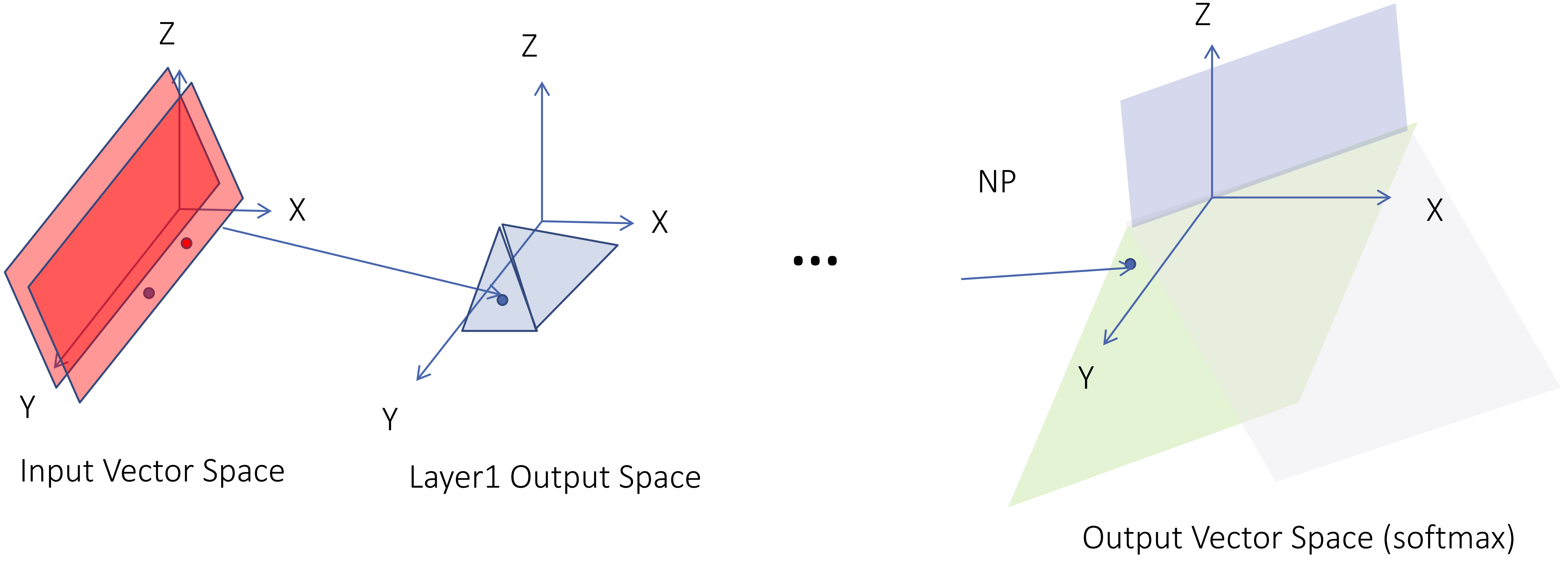
How to assign observation to the variable? – model accuracy

In neural networks,
if two observation values are dependent,
their hidden outputs generates the same output.
If the values are independent,
The vectors generate the same value.

Issues in AI and Deep Learning

In classification (determined by segmentation)

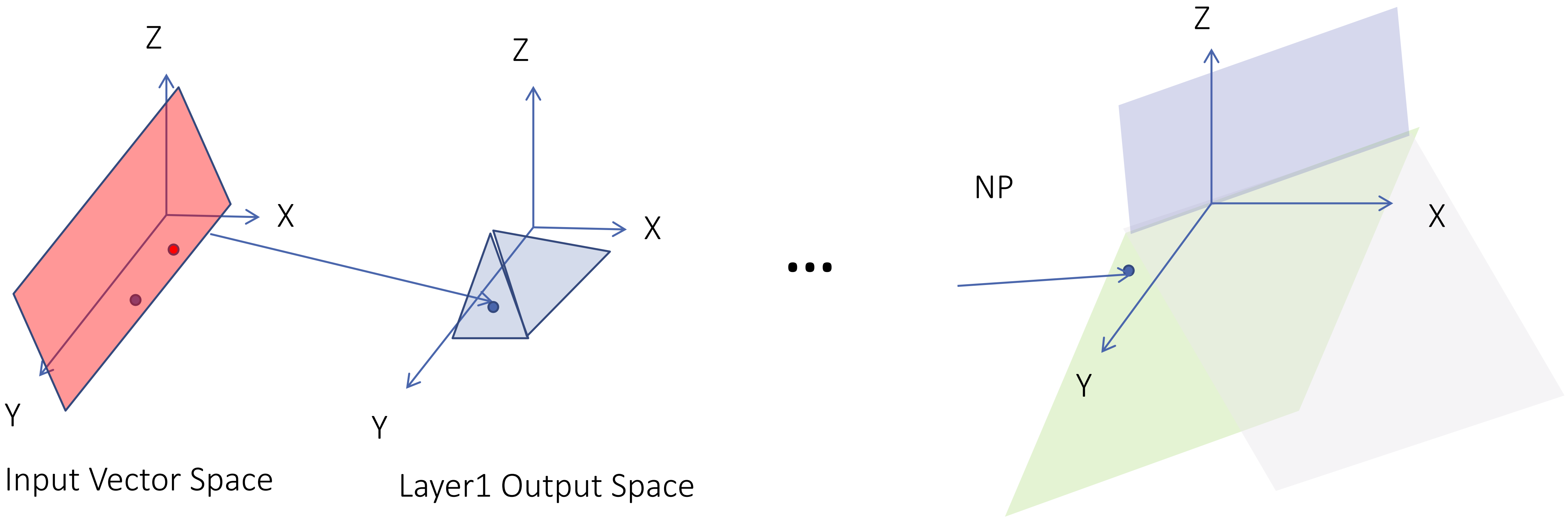
The final decision is dependent to only X



Issues in AI and Deep Learning

In regression (determined by the location on the effective region– nonzero gradient region)

The final value is dependent to only X

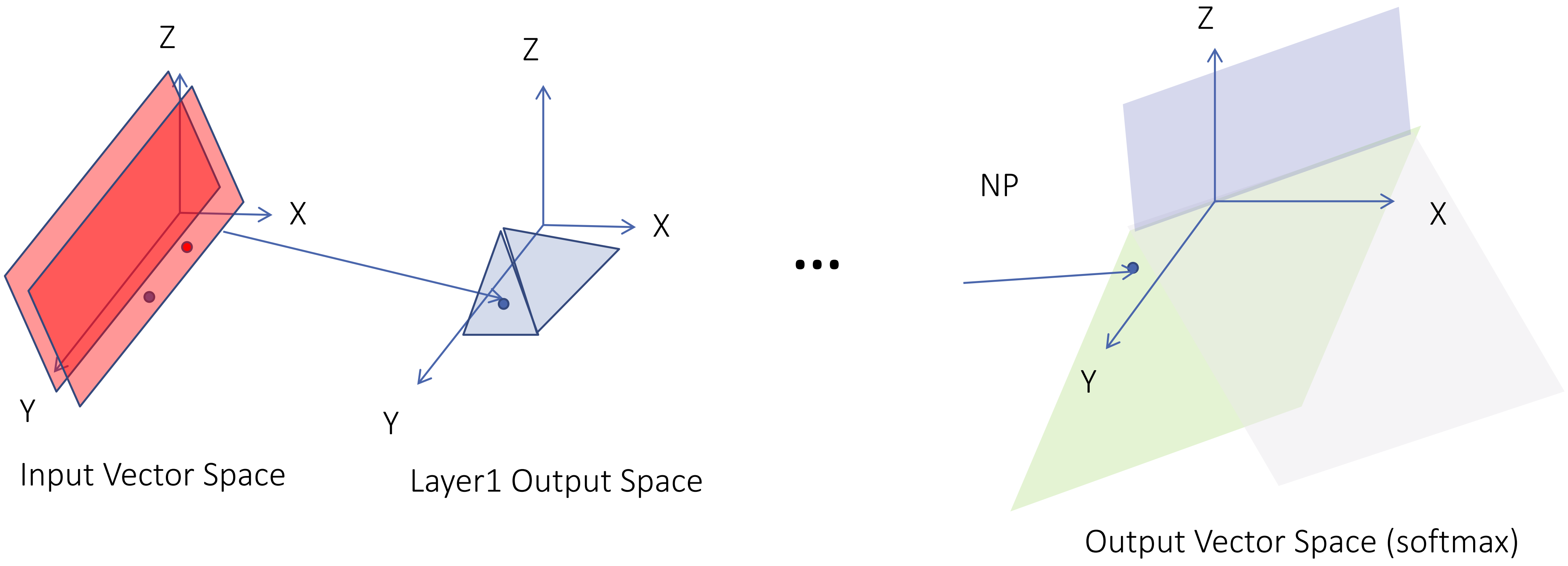


Output Vector Space (softmax)

Issues in AI and Deep Learning

In classification (determined by segmentation)

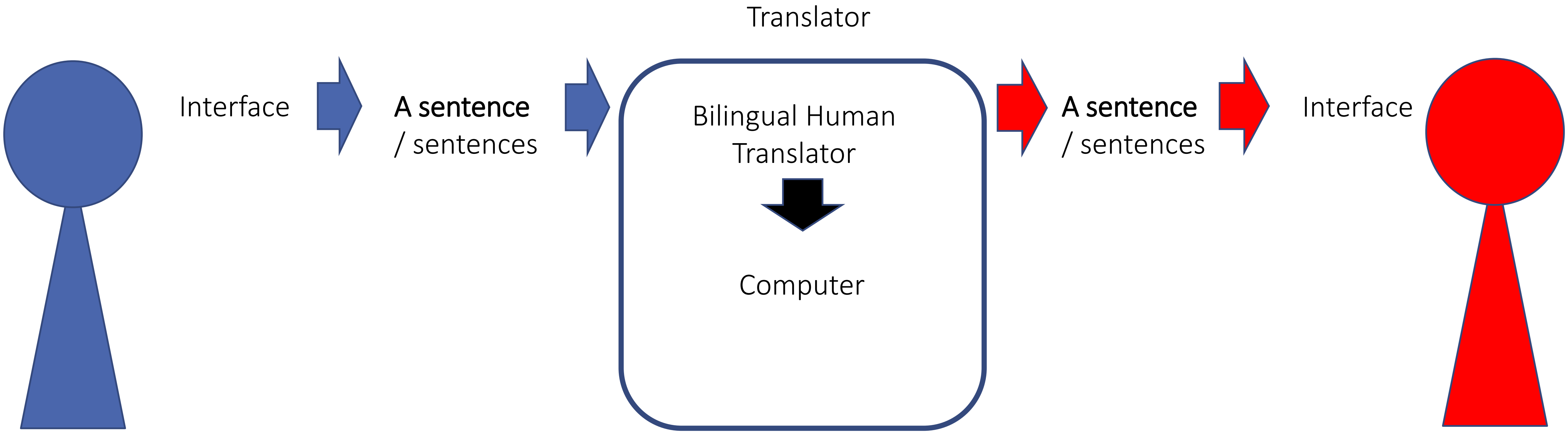
Small rotation and movement of a segment? -> changing dependency of many input vectors



Overview of Machine Translation

Overview of Machine Translation

The range of translation to be discussed in this tutorial



Overview of Machine Translation

How to build a translator?

Simplified problem definition used in the current academic community

-Input: a source sentence

-Output: a target sentence

-To build: f(source) = target

How to build “f”?

How to model “f”?

Overview of Machine Translation

Save the mapping between two sentences in computer.

If the source is matching to a saved mapping, translate it

나는 사과 먹고 싶어 -> I want to eat an apple.

Too many sentences!

usual number of words in simple conversation > 40,000

mean word size : 10 (actually it is close to 30)

40,000^10 ~ 10e+46 sentences

Too large model -> weak to unseen data

Overview of Machine Translation

Save the mapping between partial components, and build a translation

나 -> I
사과 -> an apple
먹 -> eat
~고 싶다-> want to

나는 사과 먹고 싶어
I 사과 먹고 싶어
I an apple 먹고 싶어
I an apple eat 고 싶어
I an apple eat want to
I want to eat an apple

We don’t need to save frequently used expressions and words repeatedly.

But.. We may ignore dependency between expressions

Overview of Machine Translation

I want to have an apple -> 나는 사과를 먹고 싶어
I want to have a car -> 나는 차를 가지고 싶어

have -> 먹
have -> 가지

Translation: I want to have a car -> 나는 차를 먹/가지고 싶어

How to select the correct expression?

This is not caused by ambiguity, but caused by losing dependency

Overview of Machine Translation

I want to have an apple -> 나는 사과를 먹고 싶어
I want to have a car -> 나는 차를 가지고 싶어

have an apple -> 사과를 먹
have a car -> 차를 가지

Translation: I want to have a car -> 나는 차를 가지고 싶어

Issue 1: How to know the dependency for an expression?
Issue 2: How to collect all expressions with their all dependent components?

Overview of Machine Translation

Rule-based machine translation

- Collect rules from corpus through algorithms or human experts.

A simple rule-based translation

- Source sentence analysis -> rule application -> reordering -> additional post processing

So many rules!!

- Collecting rules need too much costs
- Conflicts between rules

Overview of Machine Translation

I want to have an apple -> 나는 사과를 먹고 싶어

have an apple -> 사과를 먹
want to have -> 가지고 싶

Translation: I want to have an apple -> 나는 사과를 가지고/먹고 싶어

Overview of Machine Translation

Statistical machine translation (SMT)

- Managing all rules and combinations in a probabilistic model
- Rule selection completely relies on the probabilistic model

Goal of SMT ?

Selecting rules and combinations maximizing the probability of generating the target sentence

Overview of Machine Translation

$$argmax_e\, p(e|f) = argmax_e\, p(f|e)p(e)$$

f: a source sentence
e: a target sentence

Translation Model
- Probability of mapping components

Language Model
- Probability of the sentence
in the target language

Overview of Machine Translation

Probabilistic Model Representation for TM and LM

- N-gram, Bayesian Network, Markov Random Field, discriminative approaches
- SVM, Gaussian Mixtures, other classifiers..
- Hidden Markov Model, Conditional Random Field, other sequential classifiers..

Any traditional probabilistic models can be applied

A large number of categories for each variable ->
usually n-gram (fully connected graphical model with a given cardinality)

Overview of Machine Translation

Information in flat structures is insufficient

Expressions often have long distance dependency

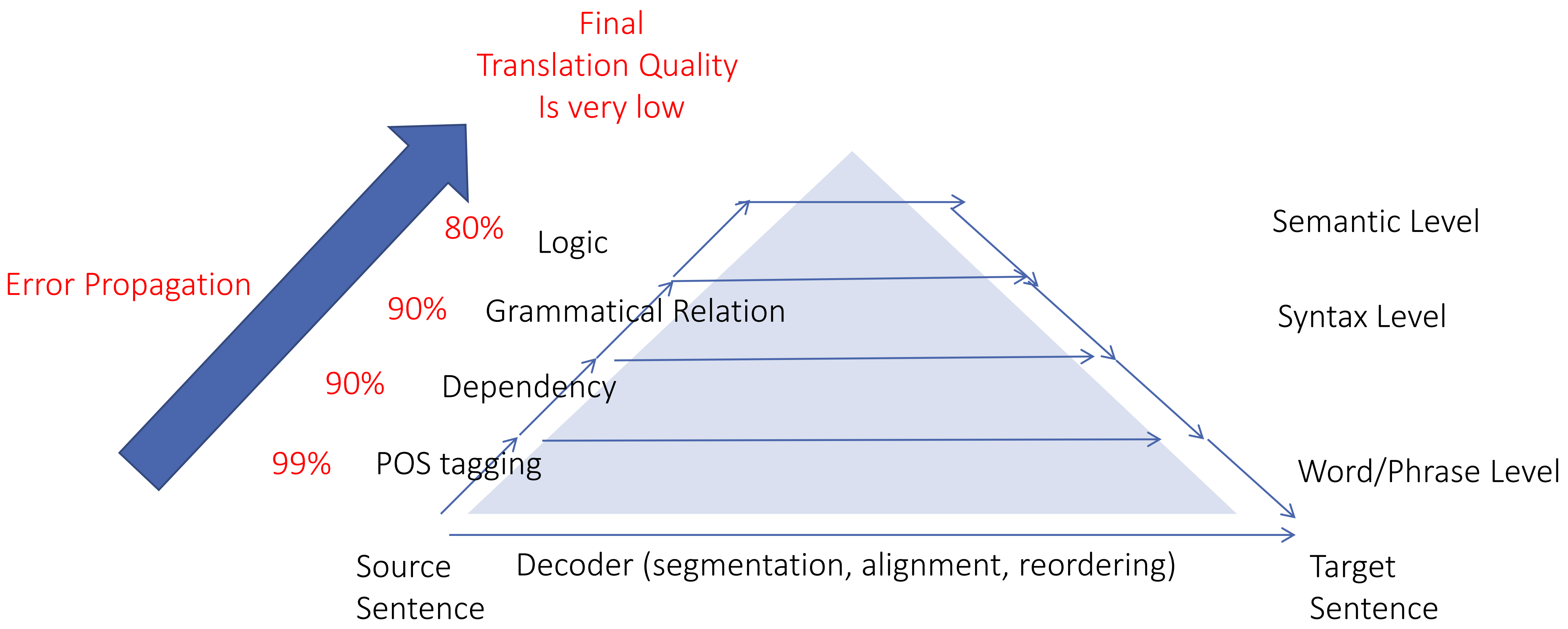
-> difficult to be detected in simple word-level decomposition of a given source sentence

Mapping patterns are often very abstract

S V O -> S O V

Syntactic and semantic analysis are required

Overview of Machine Translation



Overview of Machine Translation

Neural Machine Translation?

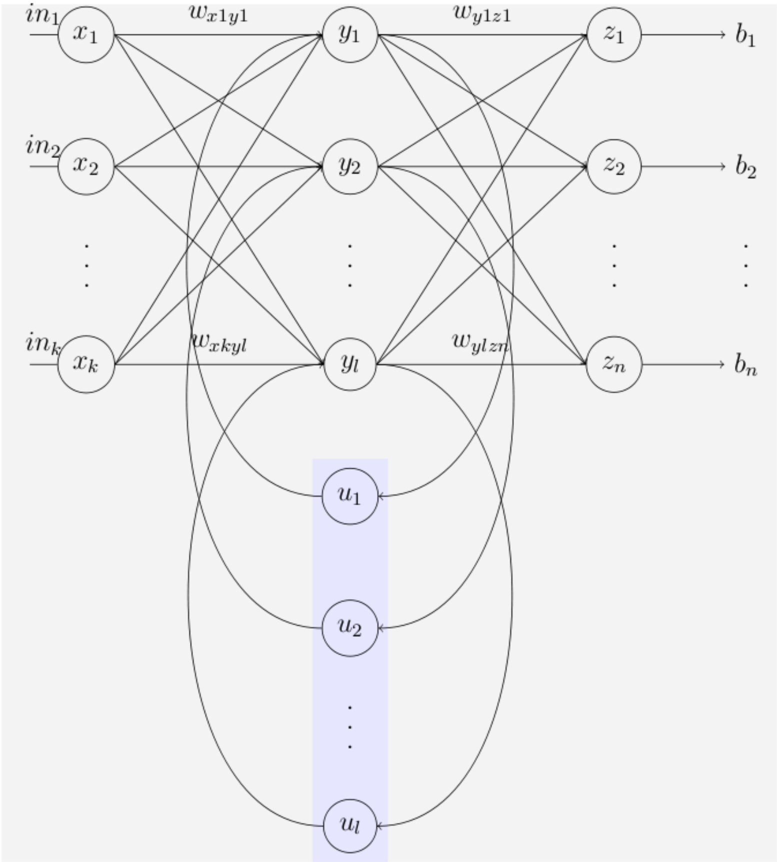
$$argmax_f \; p(e|f)$$

Learn the probability through neural networks
-> Learning conditional Language Model
-> No specific analysis and decoding process
-> every step will be trained in a neural network

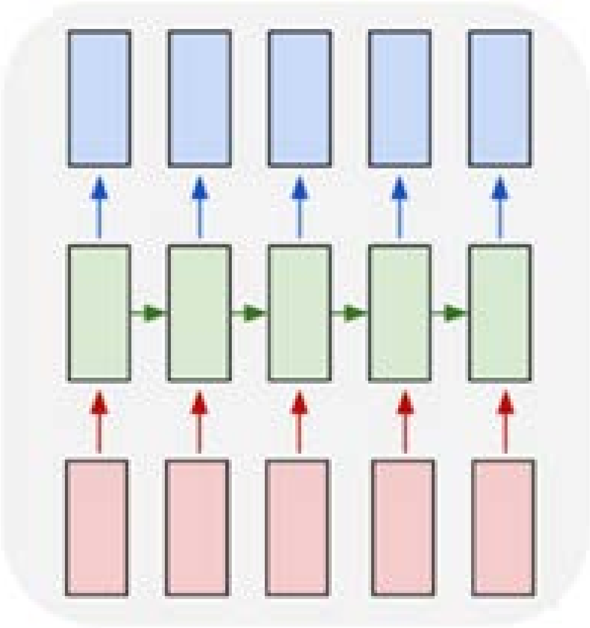
Neural Machine Translation

Neural Machine Translation

Recurrent Neural Networks (Simple Elman Network)



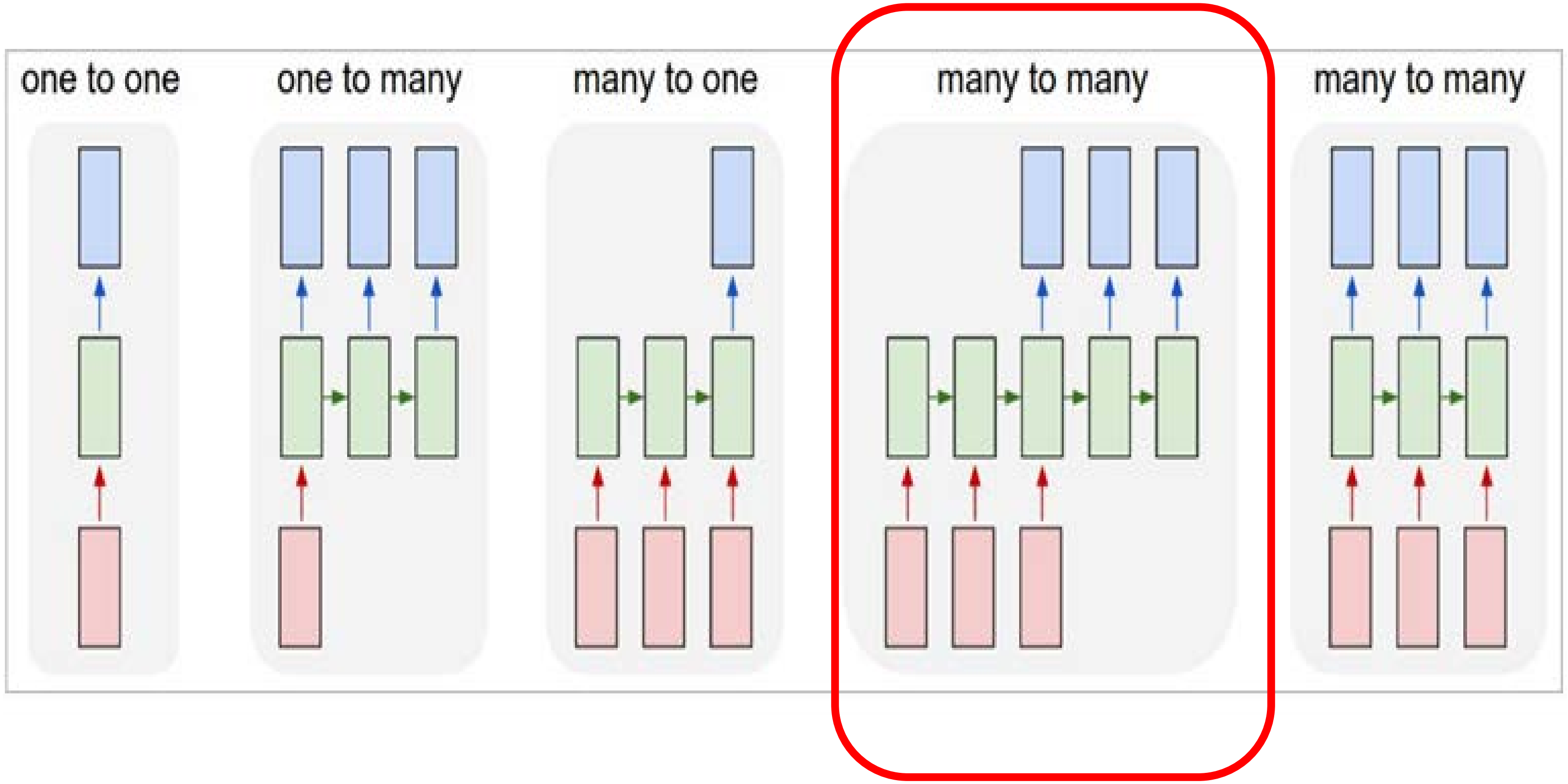
flattened by time



* Wikipedia – Recurrent neural network page

Neural Machine Translation

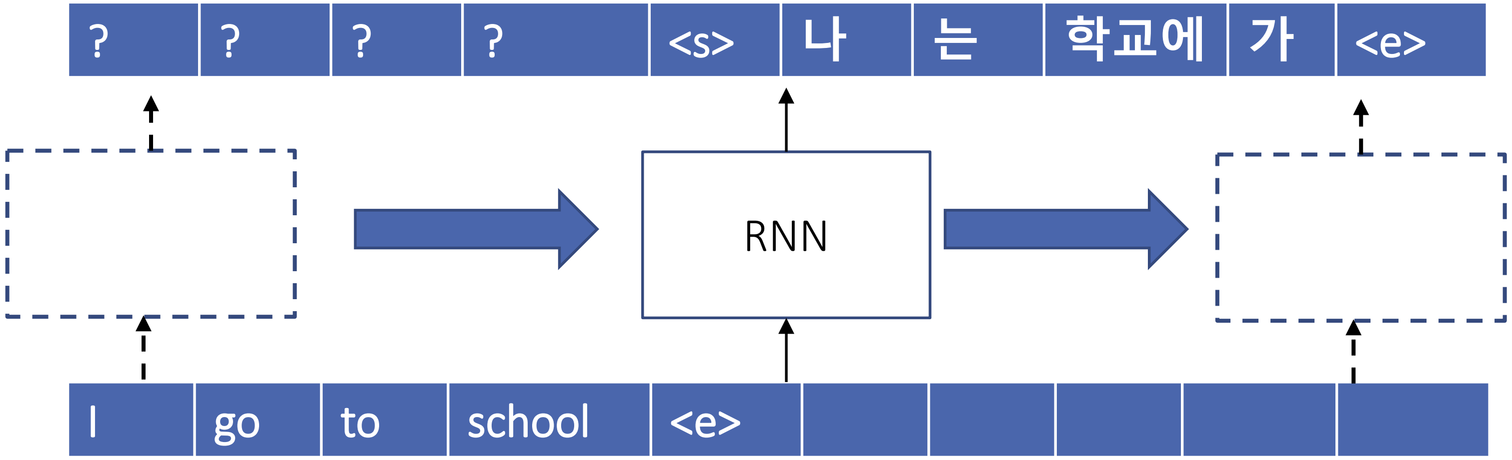
Applicable to various types of classification problems



Translation

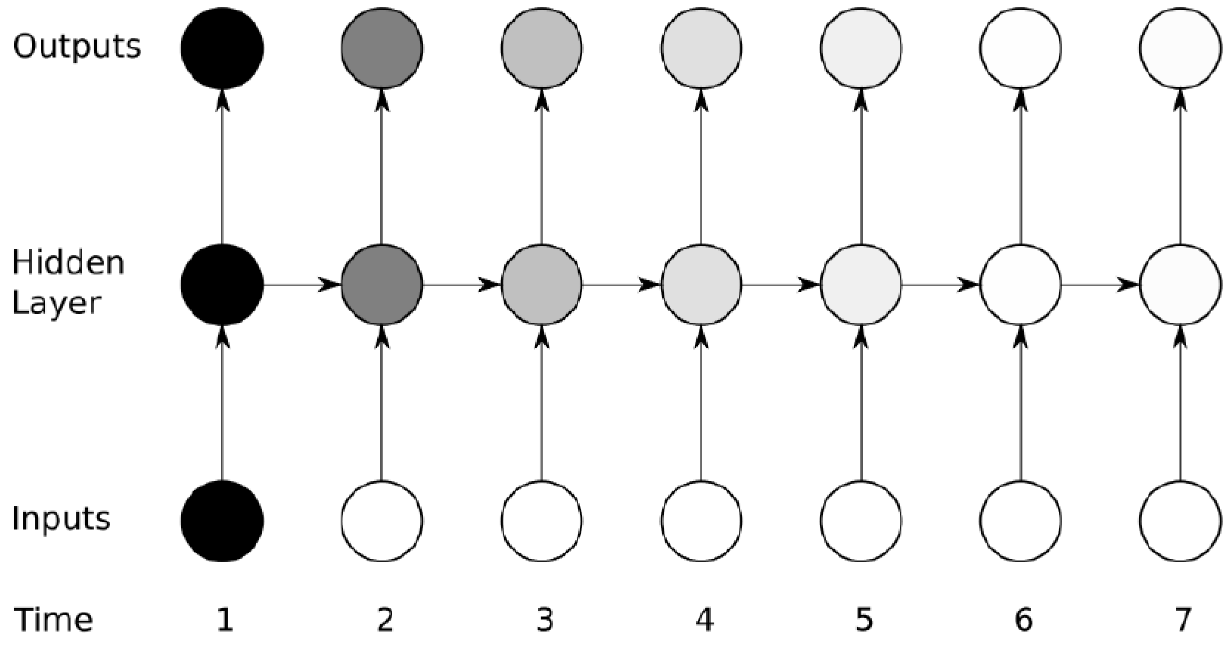
Neural Machine Translation

Recurrent Neural Networks in translation



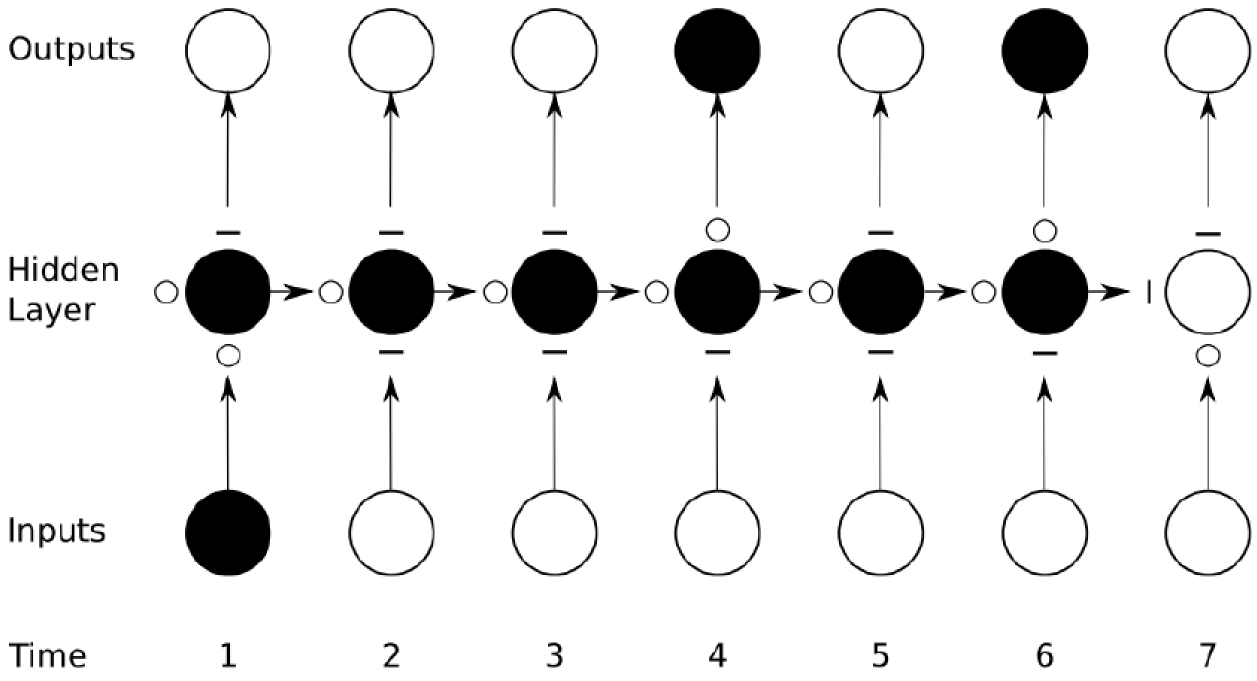
Neural Machine Translation

Recurrent Neural Networks - Gradient Vanishing over time



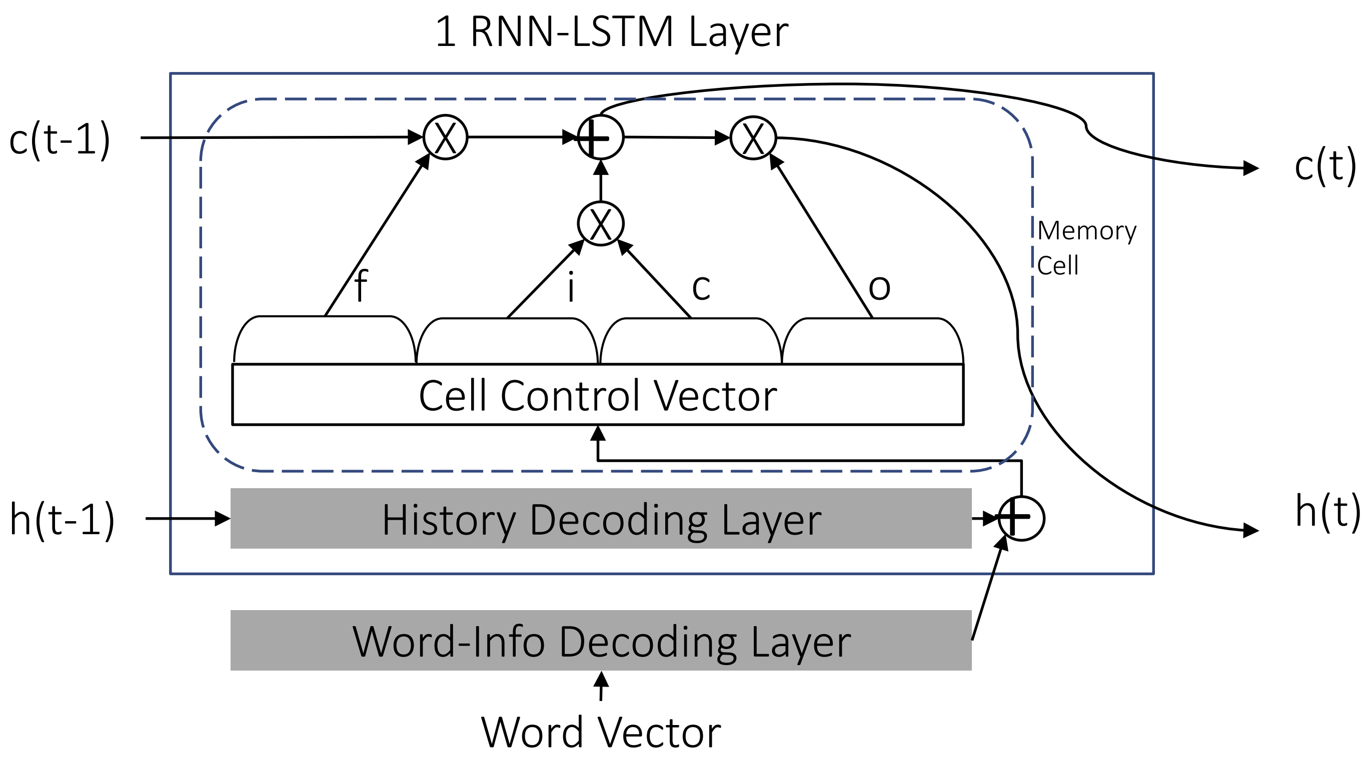
Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory



Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory– A cell

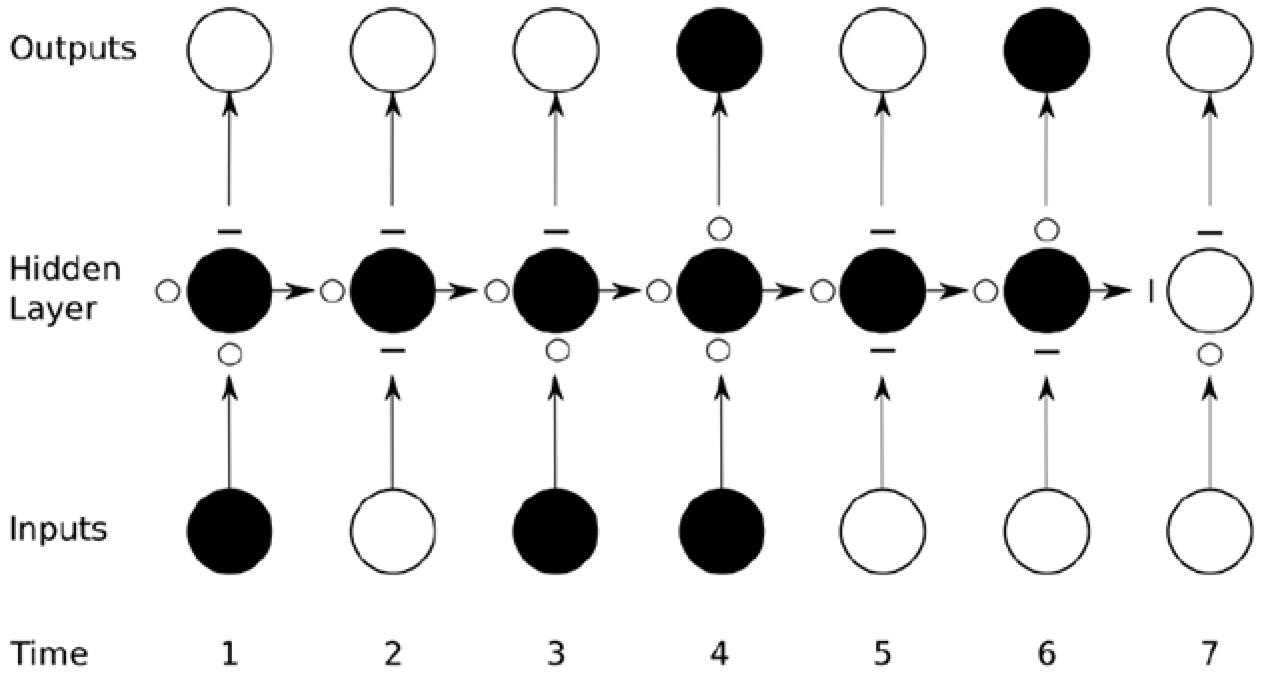


Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory– Stacked LSTM

many output values

Impact to multiple outputs



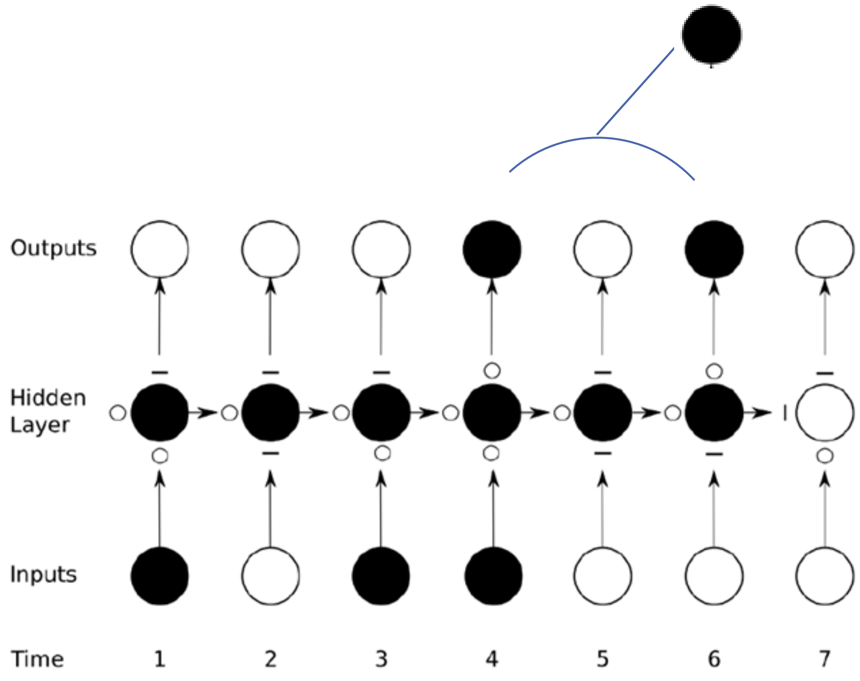
many Input values

too dense vector distribution
-> difficult to train
-> requires sufficient expression power

Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory– Stacked LSTM

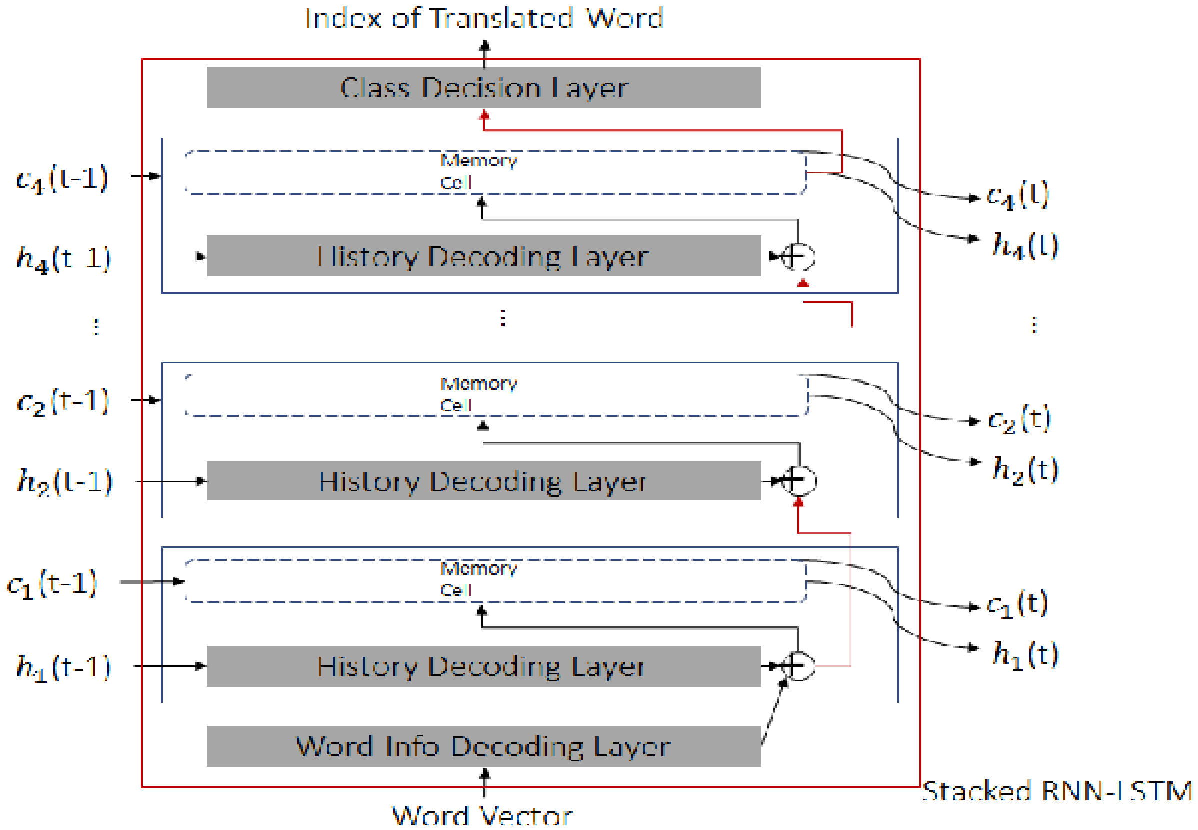
What if structural information is required?



Stacking!

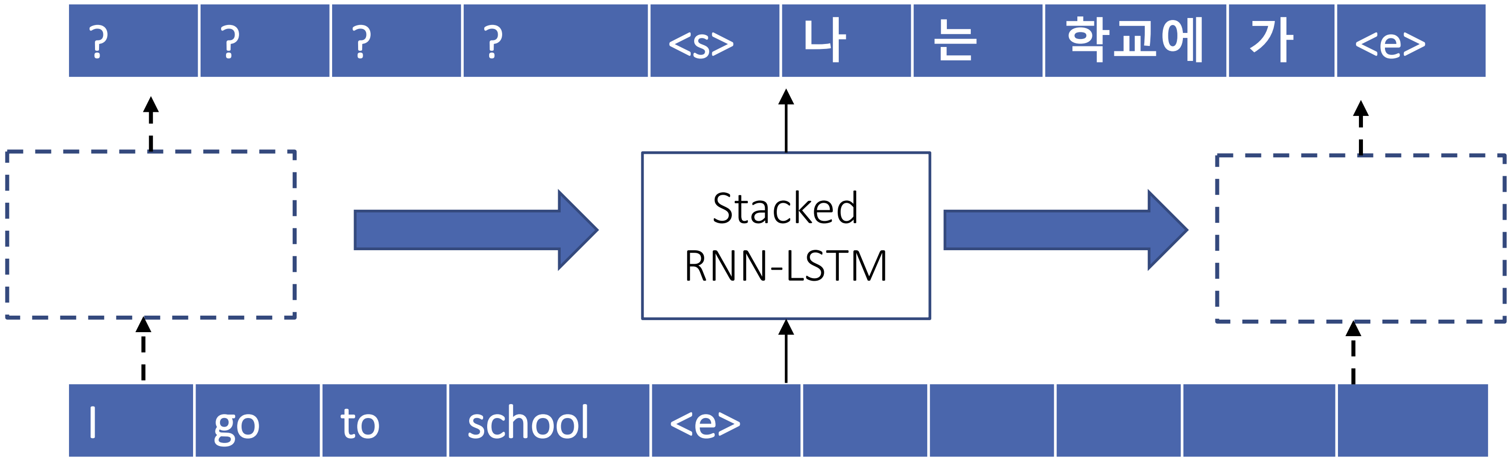
Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory– Stacked LSTM



Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory– Stacked LSTM

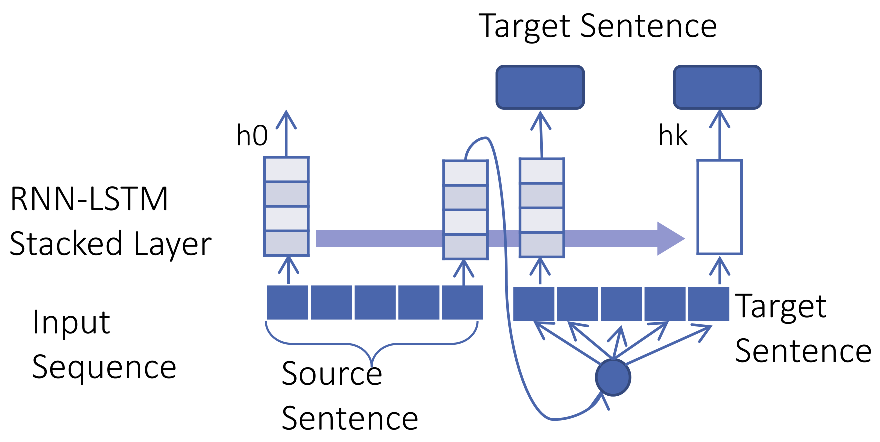


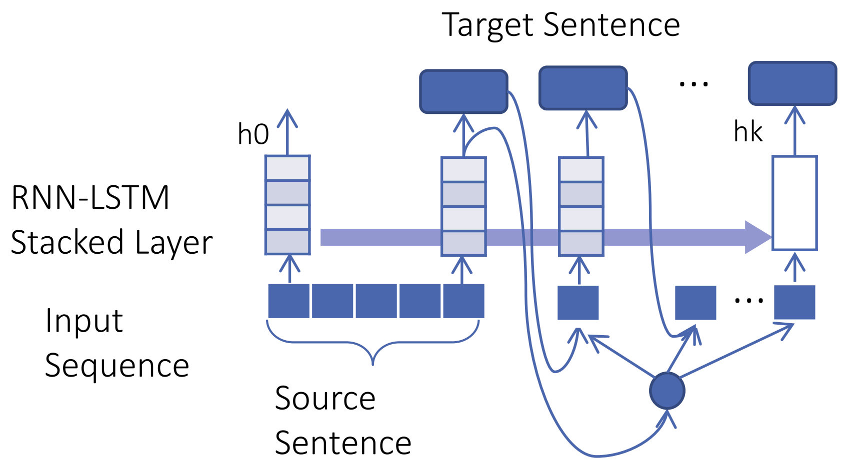
4 ~ 8 stacks
are required
for good translation
*in empirical reports

Neural Machine Translation

Recurrent Neural Networks with Long Short Term Memory– Stacked LSTM

- detailed structure





Neural Machine Translation

We saw,

-How to apply RNN, RNN with LSTM, RNN with LSTM Stacks

-Why we need complex LSTM and LSTM stacks

-How LSTM is applied to translation

Some issues to discuss..

-LSTM is proposed at about 1990, why LSTM-based translation becomes popular now? GPU, Computing Power! (Jürgen Schmidhuber, 2014, Deep Learning in Neural Networks: An Overview, IDSIA lab, Switzerland)

Neural Machine Translation

Stacked LSTM is expected to learn

structural information, long distance relation, translation equivalence, sentence decomposition

(segmentation, tagging, parsing, alignment, reordering, post processing,…, everything)

Simple LSTM can learn every information for a good translation?

No, it may represent all the conditions, but training is difficult

-> next issues in NMT: How to build networks efficiently train required information?

Advanced Techniques in Neural Machine Translation

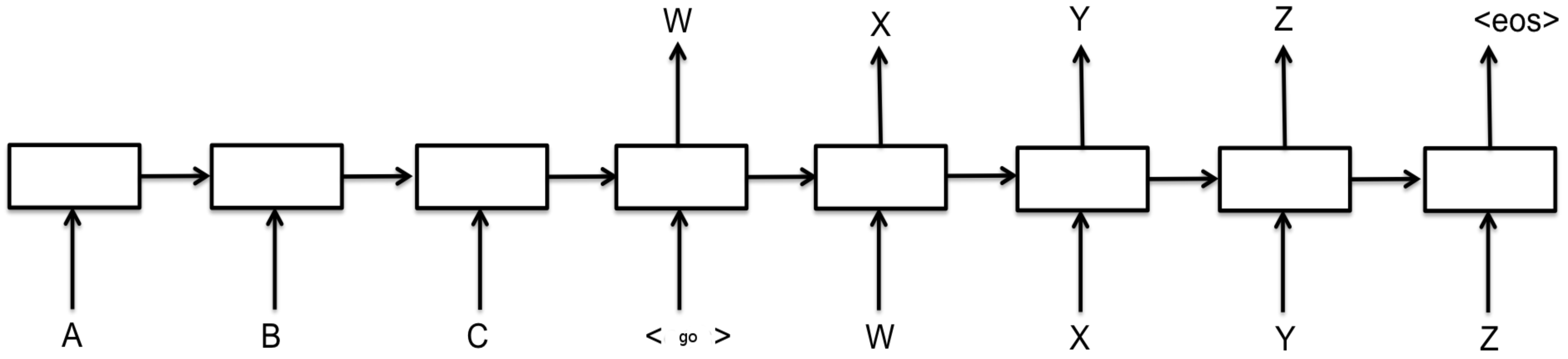
Advanced Techniques in NMT

- recurrent neural network
  - *LSTM/GRU*
- bidirectional
- attention
- syntactic guide
- direct link from input to hidden layers
- 2-dimensional grid structure
- ensemble
- explicit rare word models
- zero-Resource Training

Advanced Techniques in NMT

Recurrent Neural Network with Long Short Term Memory

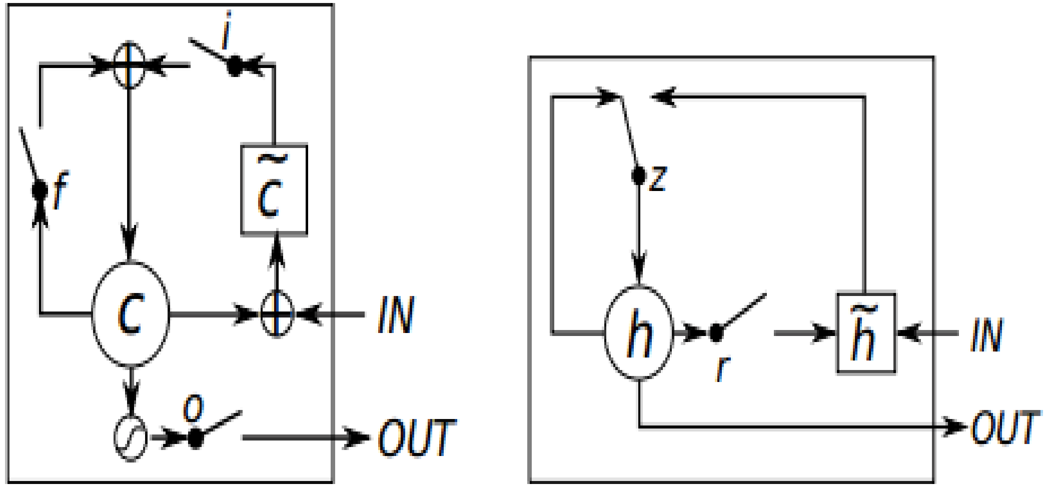
(Sutskever, 2014, Sequence to Sequence Learning with Neural Networks)



Advanced Techniques in NMT

LSTM/GRU

(Chung, 2014, Empirical evaluation of **gated recurrent** neural networks on sequence modeling)



(a) Long Short-Term Memory (b) Gated Recurrent Unit

Advanced Techniques in NMT

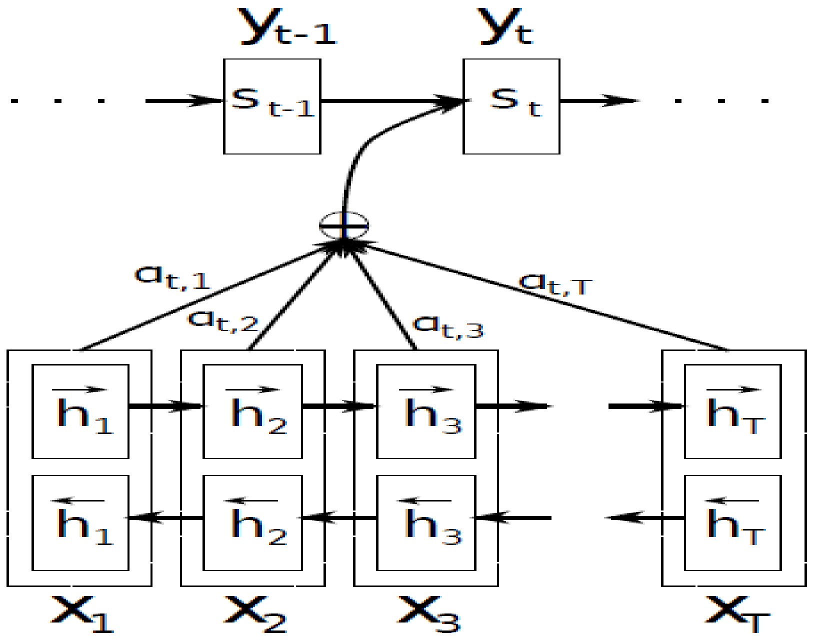
Attention and Bidirectional Model

(Bahdanau, 2015, NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE)

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j .$$

$$\alpha_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(e_{ik}\right)},$$

$$e_{ij} = a(s_{i-1}, h_j)$$



Advanced Techniques in NMT

Rare Word Modeling

(Sutskever, 2015, Addressing the Rare Word Problem in Neural Machine Translation)

en: The $\underline{unk}$ portico in $\underline{unk}$ . . .

fr: Le $\underline{unkpos}_1$ $\underline{unkpos}_{-1}$ de $\underline{unkpos}_1$ . . .

Figure 4: **Positional Unknown Model** – an example of the PosUnk model: only aligned unknown words are annotated with the $unkpos_d$ tokens.

Advanced Techniques in NMT

Syntactic Guide

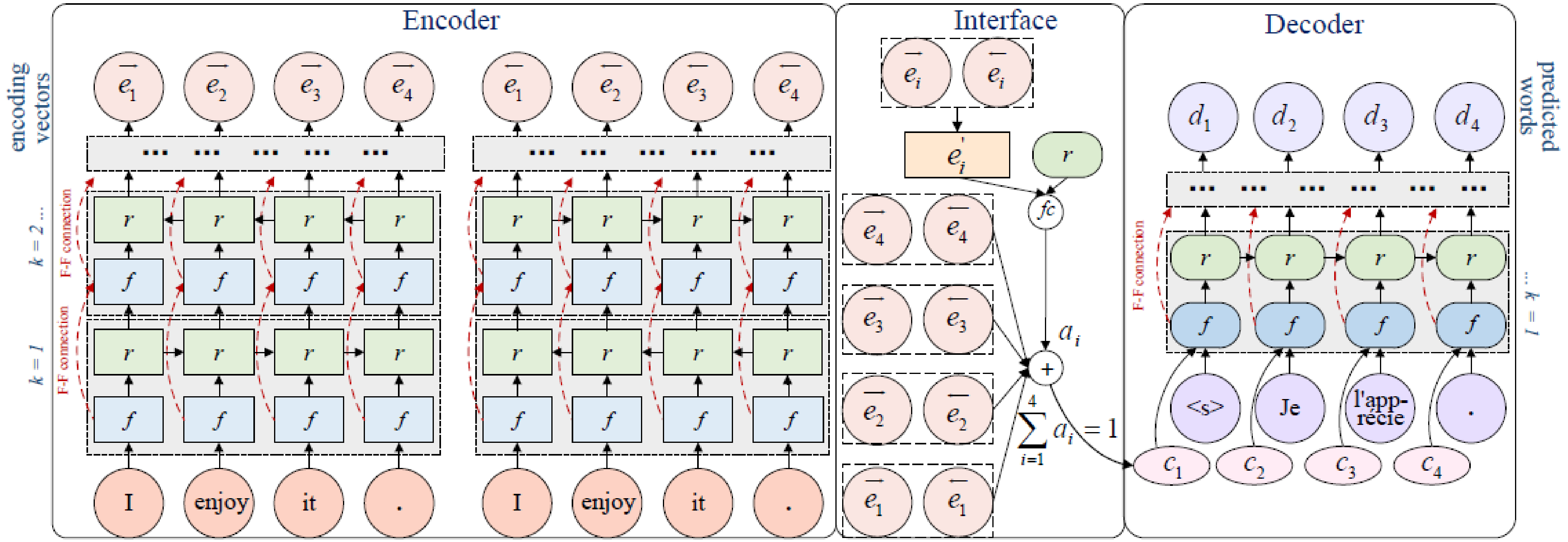
(Stahlberg, 2016, Syntactically Guided Neural Machine Translation)

$$\log P(y_t|y_1^{t-1},\mathbf{x}) = \\ \lambda_{Hiero} \log P_{Hiero}(y_t|y_1^{t-1},\mathbf{x}) + \\ \lambda_{NMT} \begin{cases} \log P_{NMT}(y_t|y_1^{t-1},\mathbf{x}) & y_t \in \Sigma_{NMT} \\ \log P_{NMT}(\mathbf{unk}|y_1^{t-1},\mathbf{x}) & y_t \notin \Sigma_{NMT} \end{cases} \tag{5}$$

Advanced Techniques in NMT

Direct Link between LSTM Stacks (Deep-Att.)

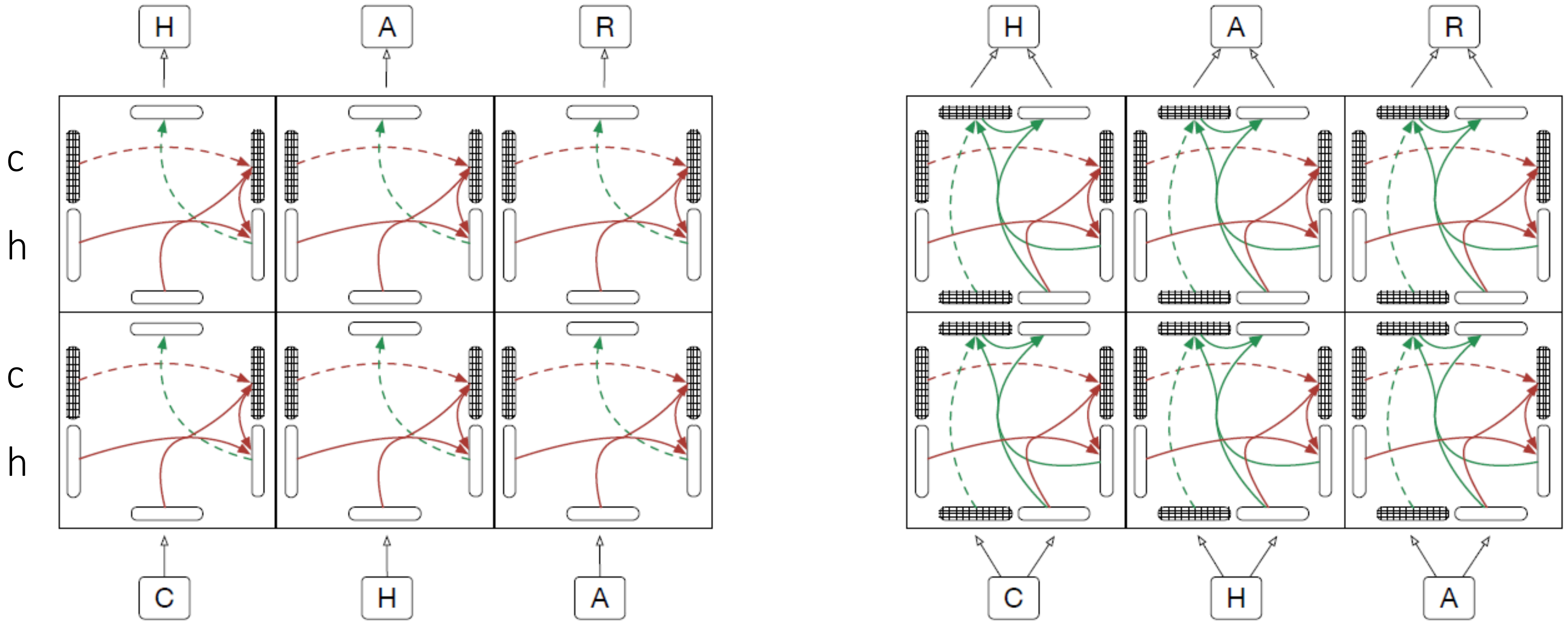
(J Zhou, 2016, Deep recurrent models with fast-forward connections for neural machine translation)



Advanced Techniques in NMT

Multidimensional LSTM

(Kalchbrenner, 2016, GRID LONG SHORT-TERM MEMORY)



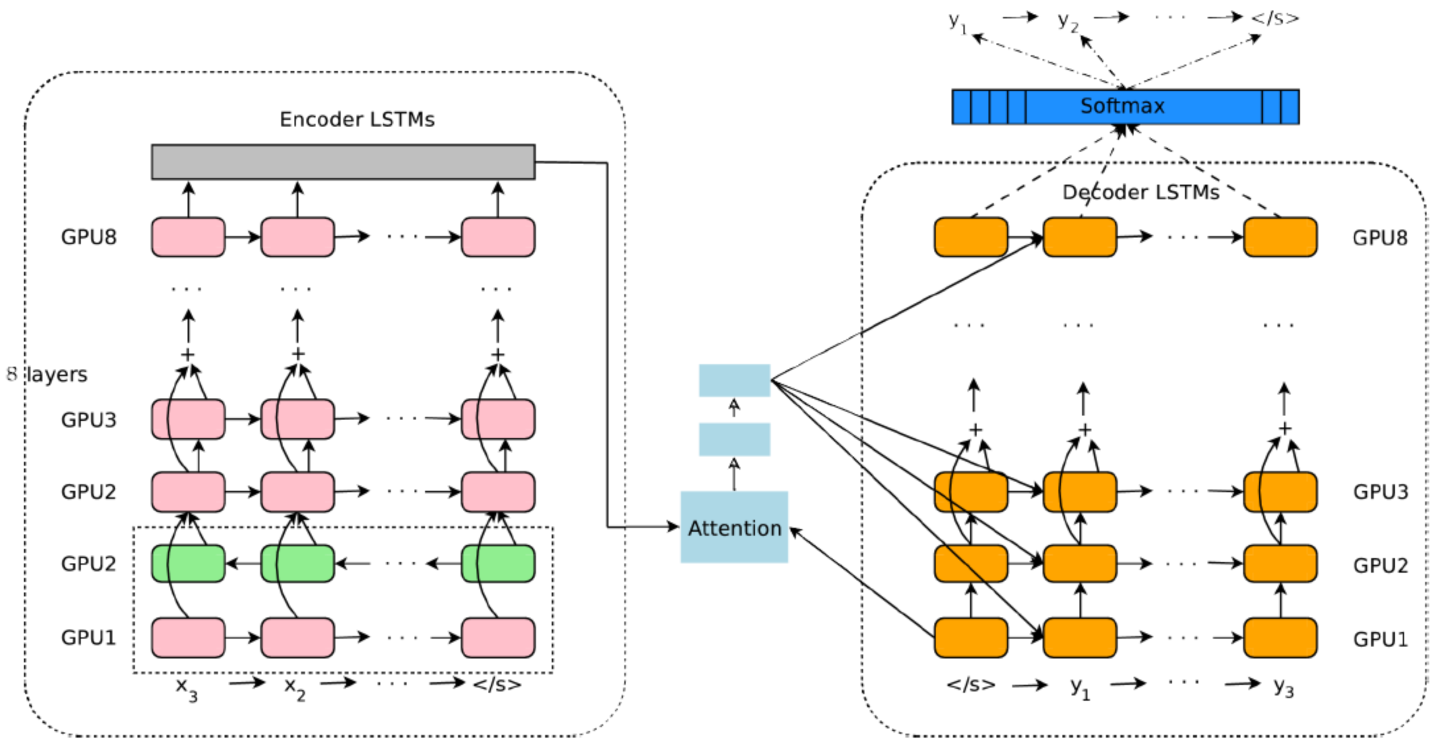
Stacked LSTM

2d Grid LSTM

Advanced Techniques in NMT

Combining most of the techniques..

(Wu, 2016, Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation)



Advanced Techniques in NMT

Zero-Resource Training (Shared Attention Model)

(Firat, 2016, **Zero-Resource Translation with Multi-Lingual Neural Machine Translation)**

- Pivot
- Shared Attention Model
- Not independent training

Issues in NMT Research

Issues in NMT Research

Google NMT Report

Table 4: Single model results on WMT En→Fr (newstest2014)

Model	BLEU	CPU decoding time per sentence (s)
Word	37.90	0.2226
Character	38.01	1.0530
WPM-8K	38.27	0.1919
WPM-16K	37.60	0.1874
WPM-32K	38.95	0.2118
Mixed Word/Character	38.39	0.2774
PBMT [15]	37.0	
LSTM (6 layers) [31]	31.5	
LSTM (6 layers + PosUnk) [31]	33.1	
Deep-Att [45]	37.7	
Deep-Att + PosUnk [45]	39.2	

Issues in NMT Research

Google NMT Report

Table 4: Single model results on WMT En→Fr (newstest2014)

Model	BLEU	CPU decoding time per sentence (s)
Word	37.90	0.2226
Character	38.01	1.0530
WPM-8K	38.27	0.1919
WPM-16K	37.60	0.1874
WPM-32K	38.95	0.2118
Mixed Word/Character	38.39	0.2774
PBMT [15]	37.0	
LSTM (6 layers) [31]	31.5	
LSTM (6 layers + PosUnk) [31]	33.1	
Deep-Att [45]	37.7	
Deep-Att + PosUnk [45]	39.2	

Table 7: Model ensemble results on WMT En→Fr (newstest2014)

Model	BLEU
WPM-32K (8 models)	40.35
RL-refined WPM-32K (8 models)	41.16
LSTM (6 layers) [31]	35.6
LSTM (6 layers + PosUnk) [31]	37.5
Deep-Att + PosUnk (8 models) [45]	40.4

Table 9: Human side-by-side evaluation scores of WMT En→Fr models.

Model	BLEU	Side-by-side averaged score
PBMT [15]	37.0	3.87
NMT before RL	40.35	4.46
NMT after RL	41.16	4.44
Human		4.82

Issues in NMT Research

Google NMT Report

Model Representation	Bidirectional (shallow layer only)	1024 nodes per layer
	Simple attention	1024 nodes per layer
	Direct link (input to LSTM stacks)	1024 nodes per layer
Optimization	Stochastic Gradient Descent/Adam mixture	
	Gradient clipping	
	Uniform weight initialization	
	Asynchronous parallel computation of gradients	
	Dropout	
	Quantization	
Translation	Beam Search	
	Postprocessing Model (reinforcement learning)	Explicit model
	Rare word replacement (target side)	Explicit model

Issues in NMT Research

Google NMT Report

Training Data Set (En-Fr)	internal set (3.6G ~36G sent.) ?
	WMT14 (36Mset.)
Hardware	12 node cluster (8 GPUs per node)
	Nvidia K80 (24G)
	Tensor Processing Unit ?
Training Time	6 days

Issues in NMT Research

Following up state-of-the-art of NMT -> GPU Clusters

For one best performance validation

Google : 6 days

Single titan X : 96 (GPUs) x 8 (ensembles) x 6 (days) = 4608 days (23 years)

May be overestimated in terms of speed improvement by parallelism

Let's assume that ?? is just 2 (Not likely)

Then 96 days

16 ~ 768 times faster

What if they use TPU in training?

160 ~ 7680 times faster

Summary

We saw,

- Properties of AI and Deep learning
- Machine translation history
- basic NMT
- The latest NMT techniques

Next NMT issues?

- efficient network structures in training
- reducing training speed (parallel processing, HW/SW, architecture...)

Google NMT
- Huge computing power is required (20M ~ sentences, En-Fr)
- at least 8 GPU machine is recommended