ML Methods:
Objectives & Paradigms

Web Mining & Retrieval, a.a. 2021-22

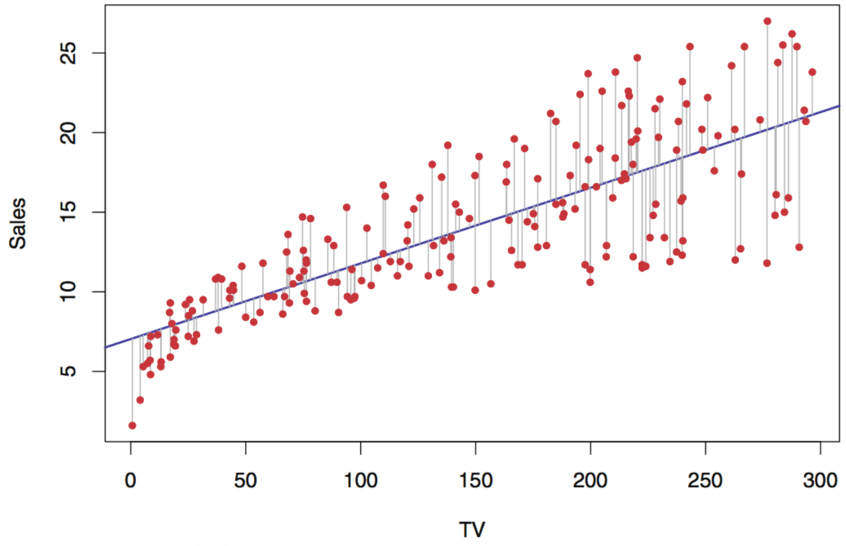
Roberto Basili

Summary

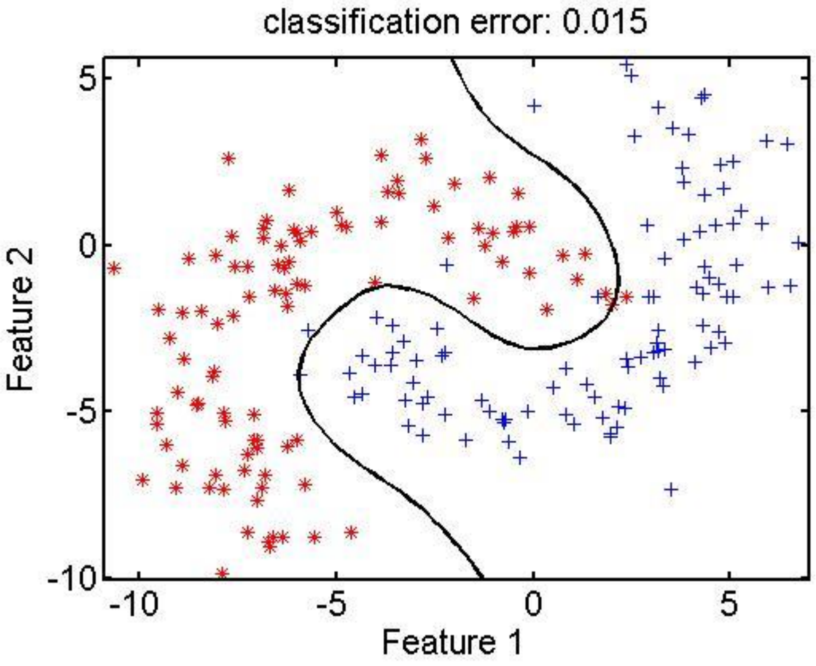
- Target problems for Machine Learning
- Geometrical Paradigms
- Probabilistic Paradigms
  - Generative models
  - Applications to speech and language processing

Machine Learning: the core problems

Regression

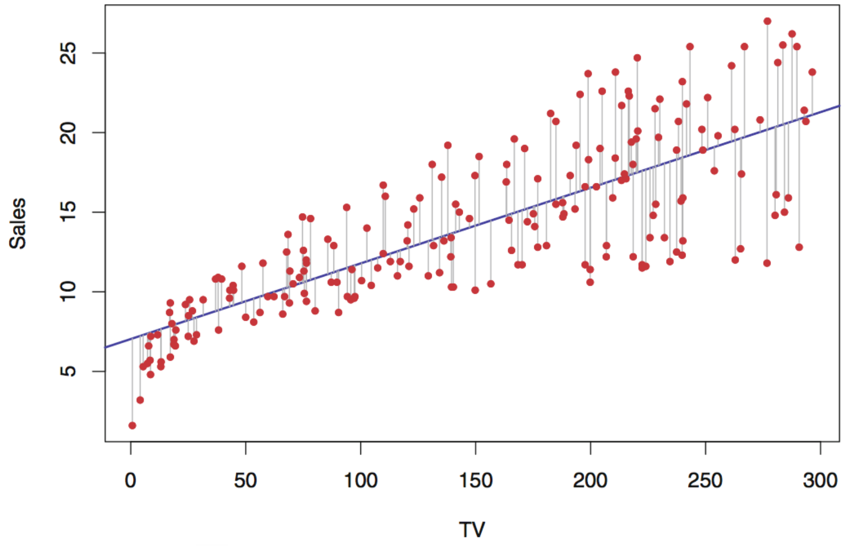


Classification

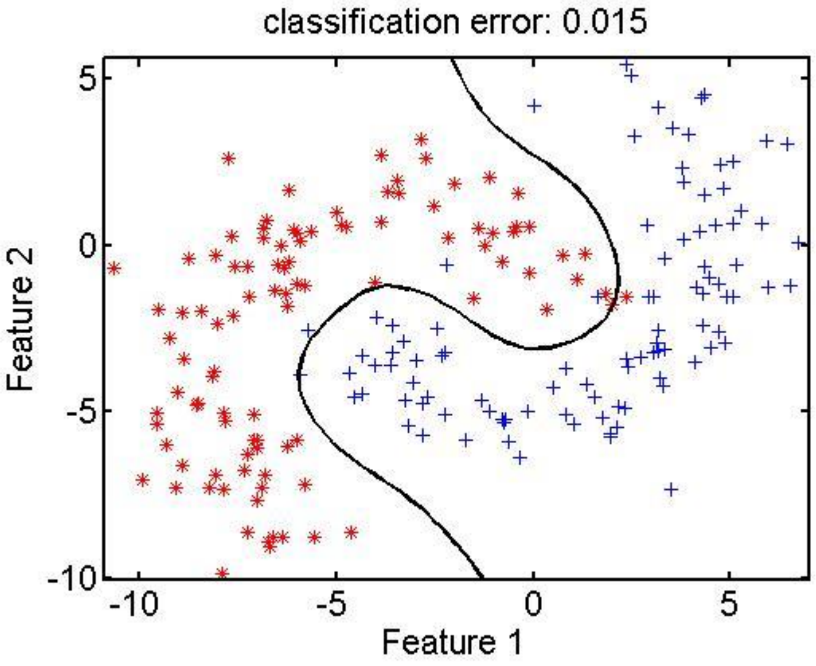


Machine Learning: the core problems

Regression



Classification



Machine Learning: the core problems

Regression

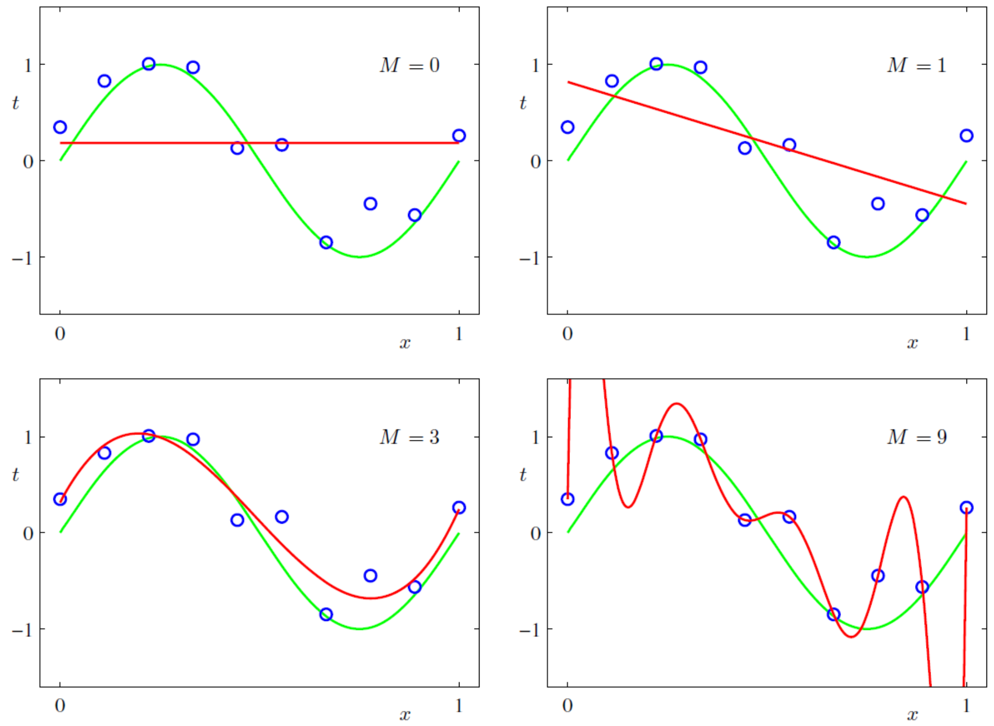
- Given a set of examples of a target function *f(.)*
- $x_1, \dots., x_k$ with $y_i=f(x_i)$ known for every i
- Define a function *h(.)* such that:
  - $h(x_i) = y_i = f(x_i) \quad \forall i$
  - $h(x) \approx f(x)$ *elsewhere*

Classification

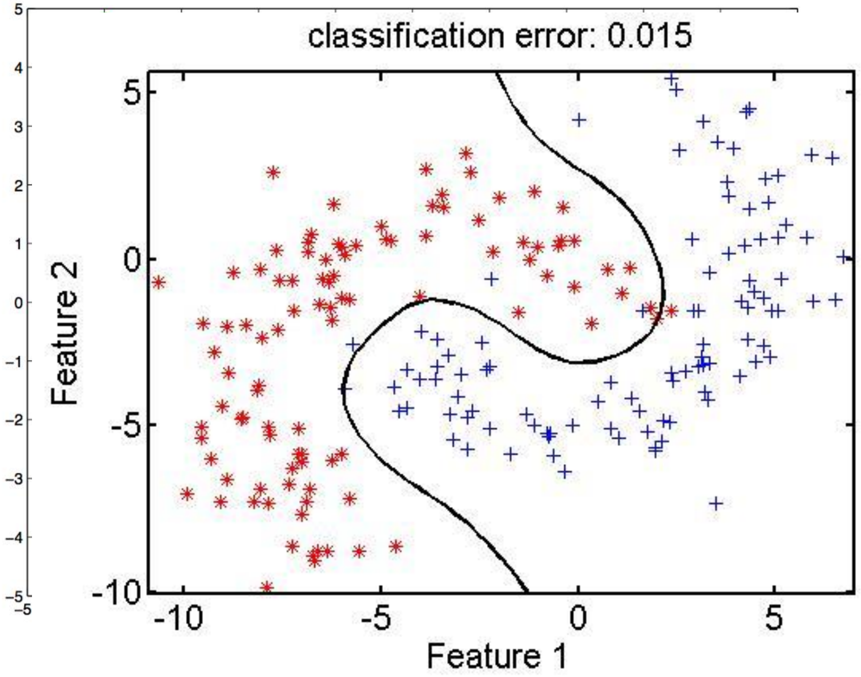
- Given *n* classes $C_1, \dots C_n$ and a given number of instances $x_1, \dots., x_k$ whose classification $y_1, \dots., y_k$ is known
- Define the class membership function *h(.)* such that
  - $h(x_i) = y_i \quad \forall i=1, \dots, k$
  - $h(x) \triangleq C_i$ such that (by definition) $x \in C_i$ for all other x

Machine Learning: la scelta delle funzioni

Regression



Classification

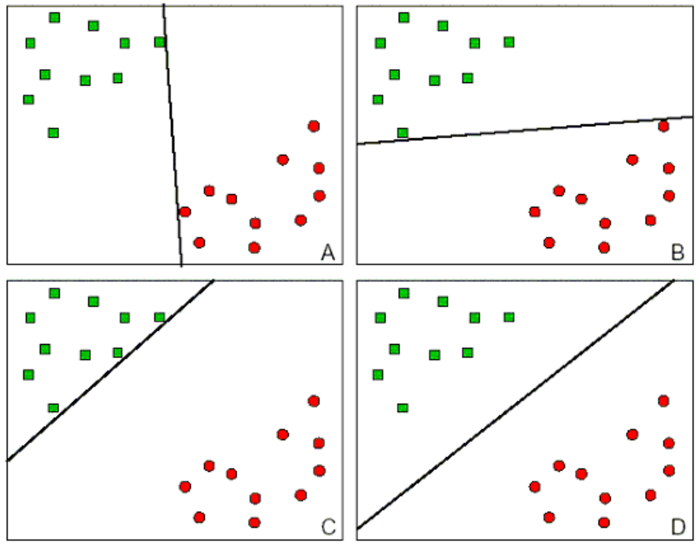


Paradigms for Model Selection

- Model Selection depends on the choice of:
  - (**Model Family Selection**) a class/family of functions (e.g. polynomials of degree *n*)
  - (**Model parametrization**). Selection/Estimation of the parameters suitable for defining the optimal decision function
    - Definition of the notion of optimality (e.g. *coverage* vs. accuracy)
    - Search for the optimal values of the parameters
      - Analytical forms
      - Empirical induction from the training set

Model Selection from a family of functions

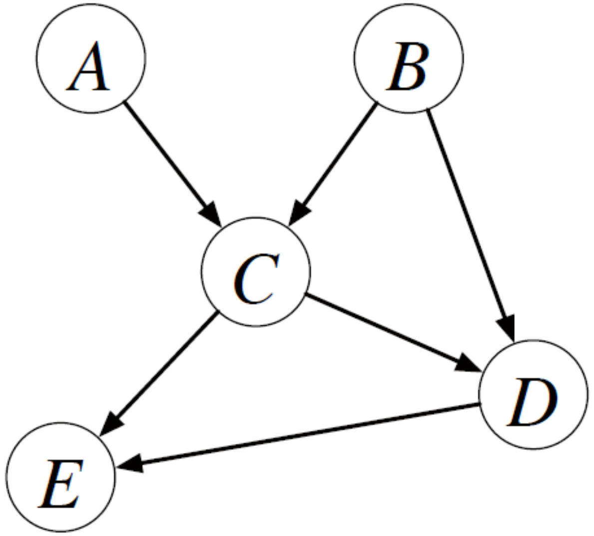
- Discriminative approaches
  - Linear models
  - h(x) = sign(**W** · x + b)



  - Probabilstic approaches
    - Estimates of probabilities probabilità $p(\mathcal{C}_k|\mathbf{x})$ over a training set
    - Generative Model of the target task allows the application of the Bayesin inversion

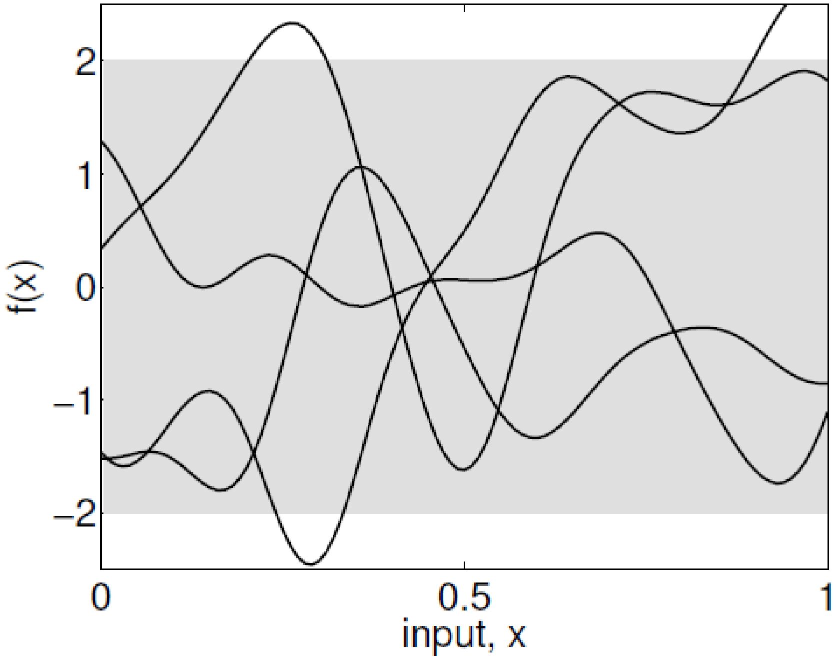
$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}.$$

Graphical Models

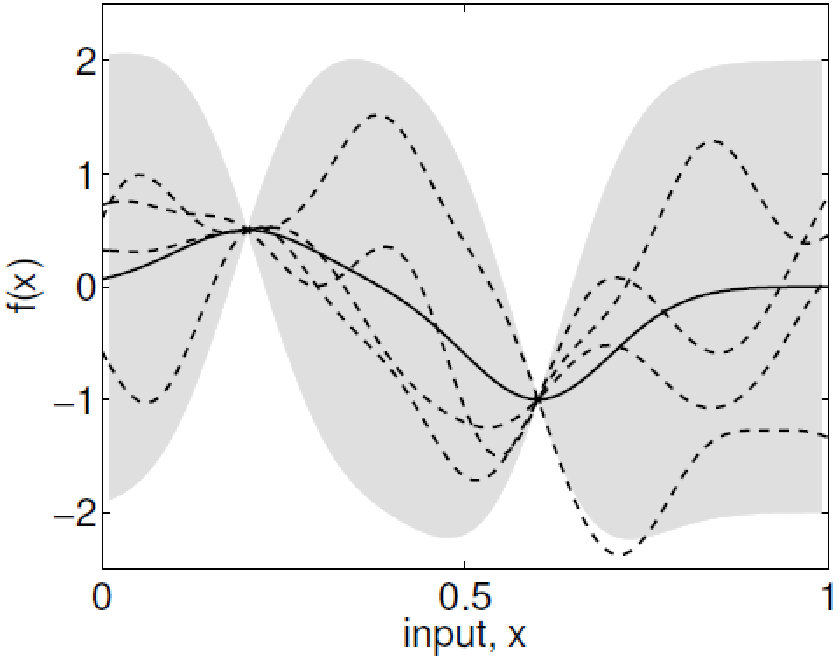


$$p(A, B, C, D, E) = p(A)p(B)p(C|A, B)p(D|B, C)p(E|C, D)$$

Bayesian & Grafical models



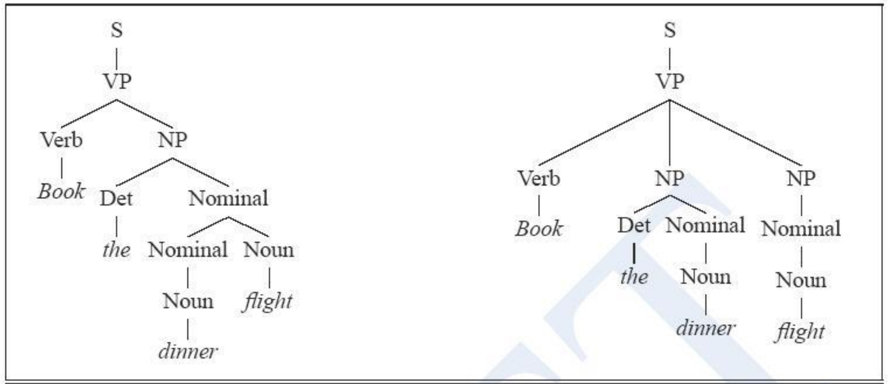
(a), prior



(b), posterior

Weighted Grammars: Languages, Syntax & Statistics

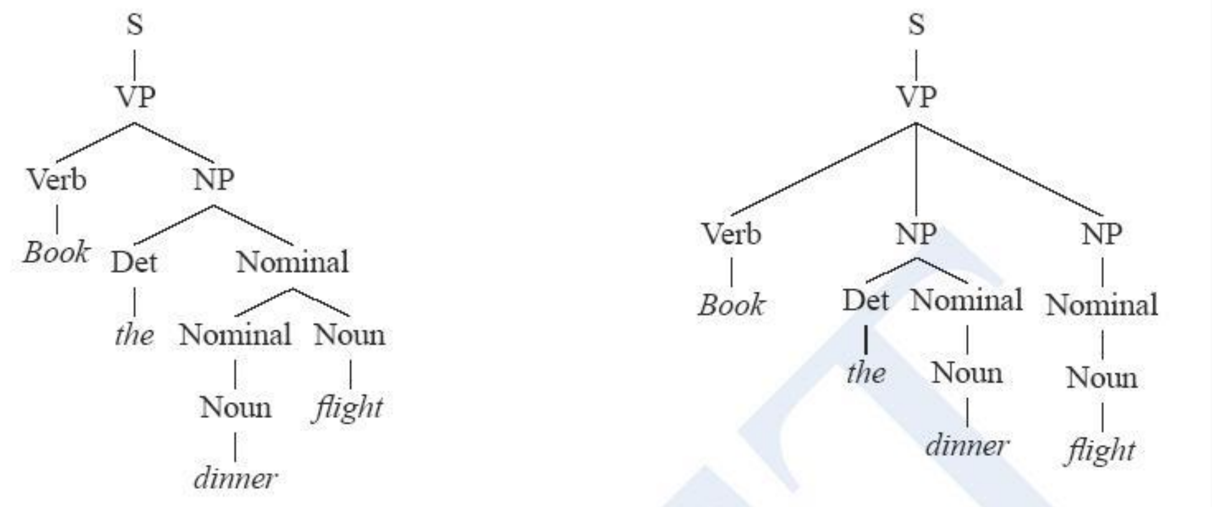
- POS tagging (Curch, 1989)
- Probabilistic Context-Free Grammars (Pereira & Schabes, 1991)
- Data Oriented Parsing (Scha, 1990)
- Stochastic Grammars (Abney, 1993)
- Lessicalizzati Modelli (C. Manning, 1995)



Rules			P	Rules			P
S	→	VP	.05	S	→	VP	.05
VP	→	Verb NP	.20	VP	→	Verb NP NP	.10
NP	→	Det Nominal	.20	NP	→	Det Nominal	.20
Nominal	→	Nominal Noun	.20	NP	→	Nominal	.15
Nominal	→	Noun	.75	Nominal	→	Noun	.75
				Nominal	→	Noun	.75
Verb	→	book	.30	Verb	→	book	.30
Det	→	the	.60	Det	→	the	.60
Noun	→	dinner	.10	Noun	→	dinner	.10
Noun	→	flights	.40	Noun	→	flights	.40

Figure 13.2 Two parse trees for an ambiguous sentence. The transitive parse (a) corresponds to the sensible meaning "Book flights that serve dinner", while the ditransitive parse (b) to the nonsensical meaning "Book flights on behalf of 'the dinner'?".

Weighted Grammars, between Syntax & Statistics

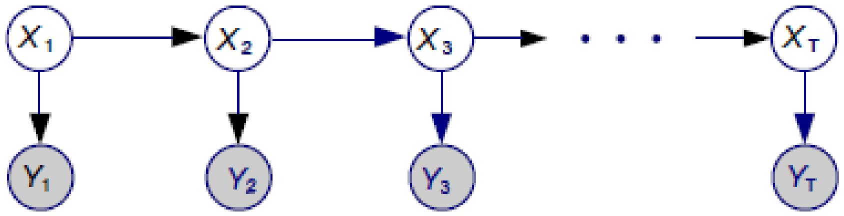


Rules			P
S	→	VP	.05
VP	→	Verb NP	.20
NP	→	Det Nominal	.20
Nominal	→	Nominal Noun	.20
Nominal	→	Noun	.75
Verb	→	book	.30
Det	→	the	.60
Noun	→	dinner	.10
Noun	→	flights	.40

Rules			P
S	→	VP	.05
VP	→	Verb NP NP	.10
NP	→	Det Nominal	.20
NP	→	Nominal	.15
Nominal	→	Noun	.75
Nominal	→	Noun	.75
Verb	→	book	.30
Det	→	the	.60
Noun	→	dinner	.10
Noun	→	flights	.40

Figure 13.2 Two parse trees for an ambiguous sentence, The transitive parse (a) corresponds to the sensible meaning "Book flights that serve dinner", while the ditransitive parse (b) to the nonsensical meaning "Book flights on behalf of 'the dinner'?".

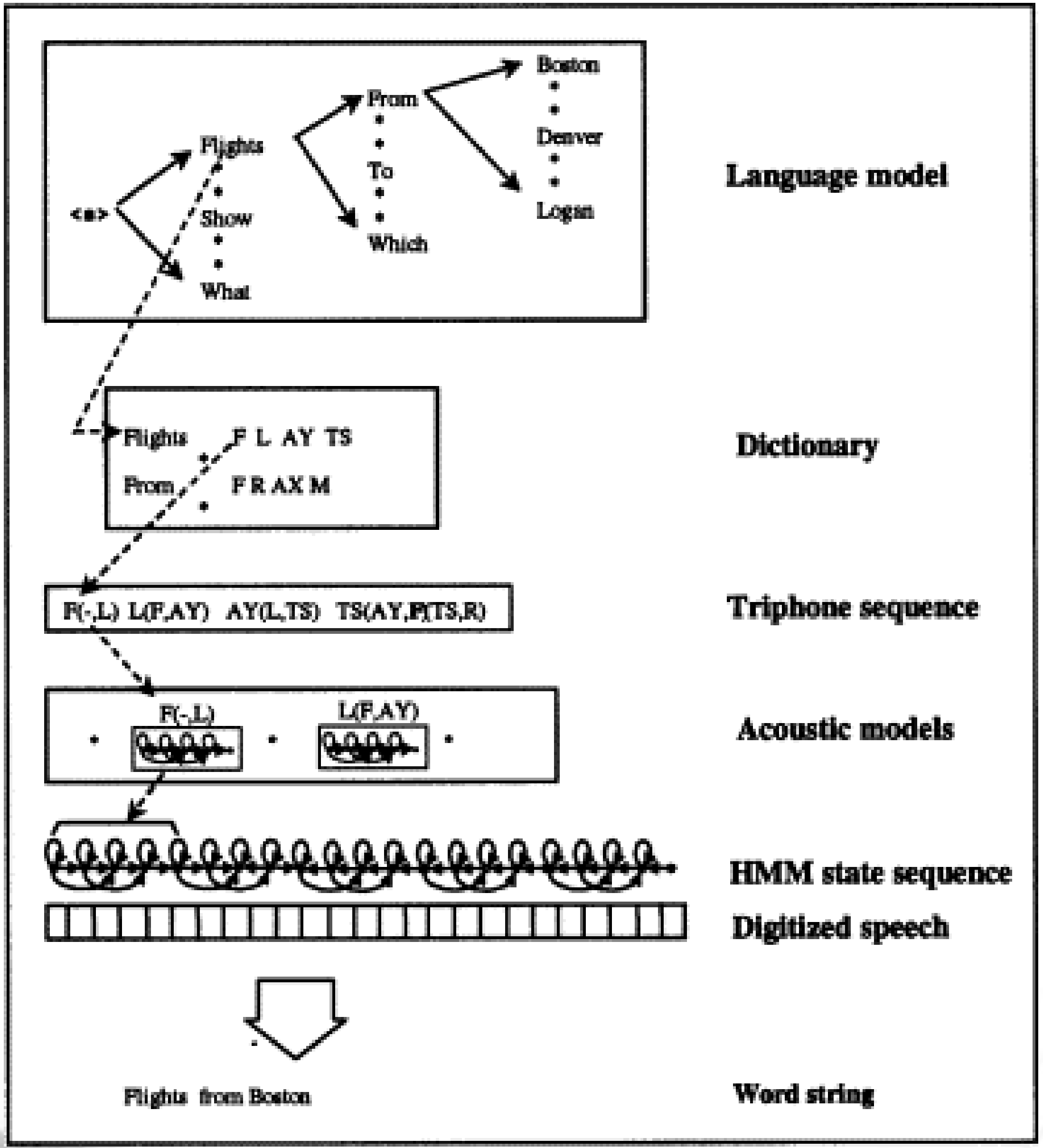
Hidden Markov Models



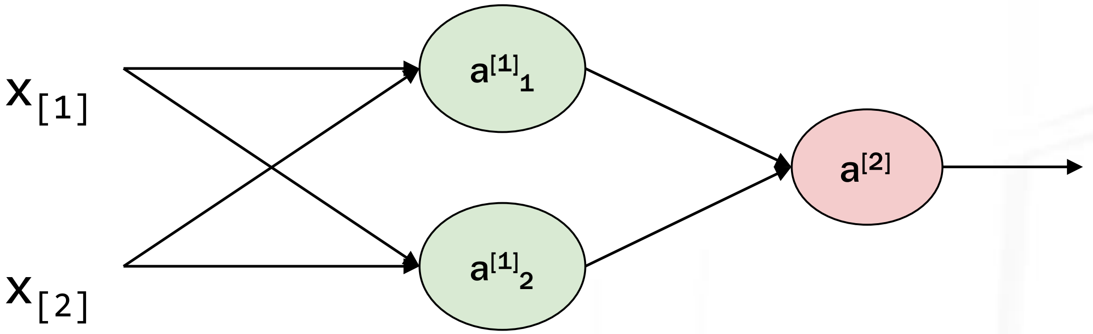
$$p(X_{1,\ldots,T}, Y_{1,\ldots,T}) = p(X_1)p(Y_1|X_1)\prod_{t=2}^{T}[p(X_t|X_{t-1})p(Y_t|X_t)]$$

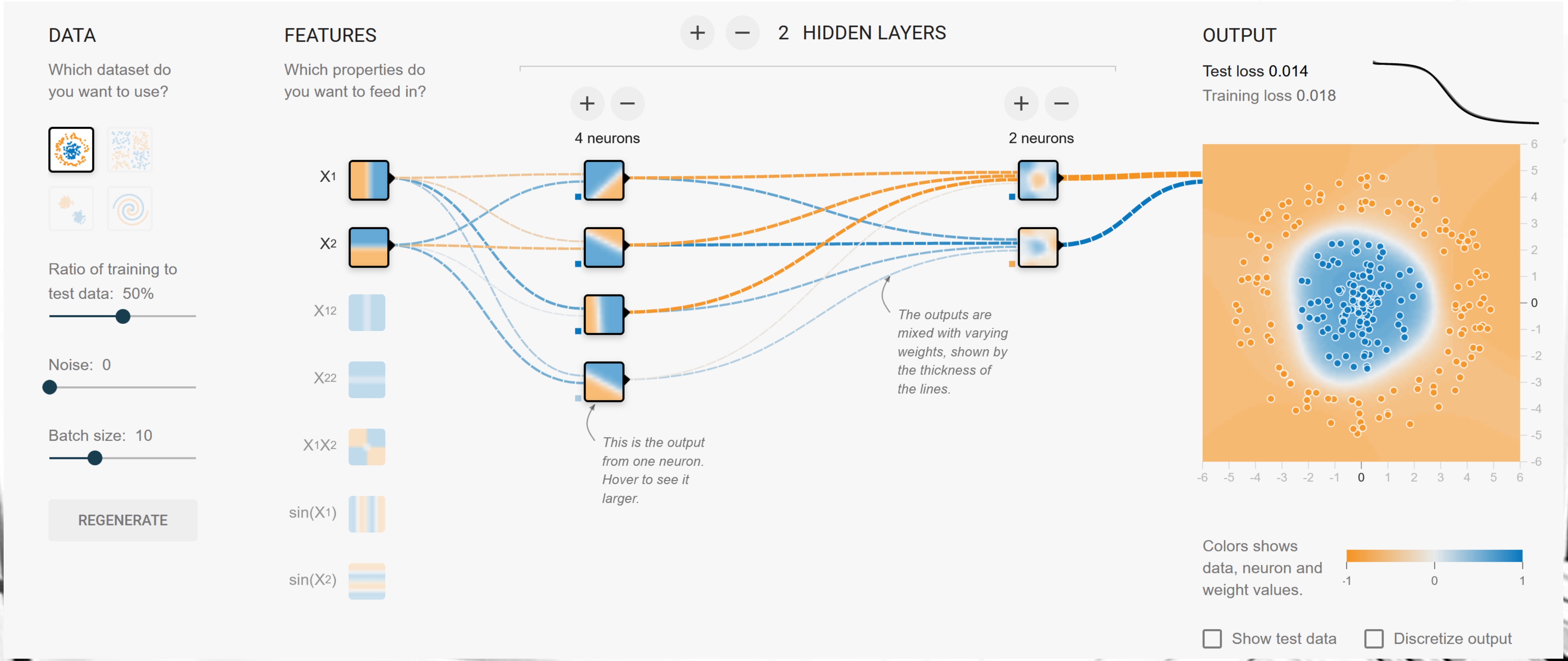
- States = Categories/Concepts/Properties
- Observations: (sequences of) symbols characterizing a given language
- Emissions (of symbols by States) vs. Transitions (between states)
- Applications:
  - *Speech Recognition* (symbols: phonems, states: segments of audio signal)
  - *POS tagging* (symbols: words, states: grammatical categories, i.e. POS tags)

HMM for Automatic Speech Recognition



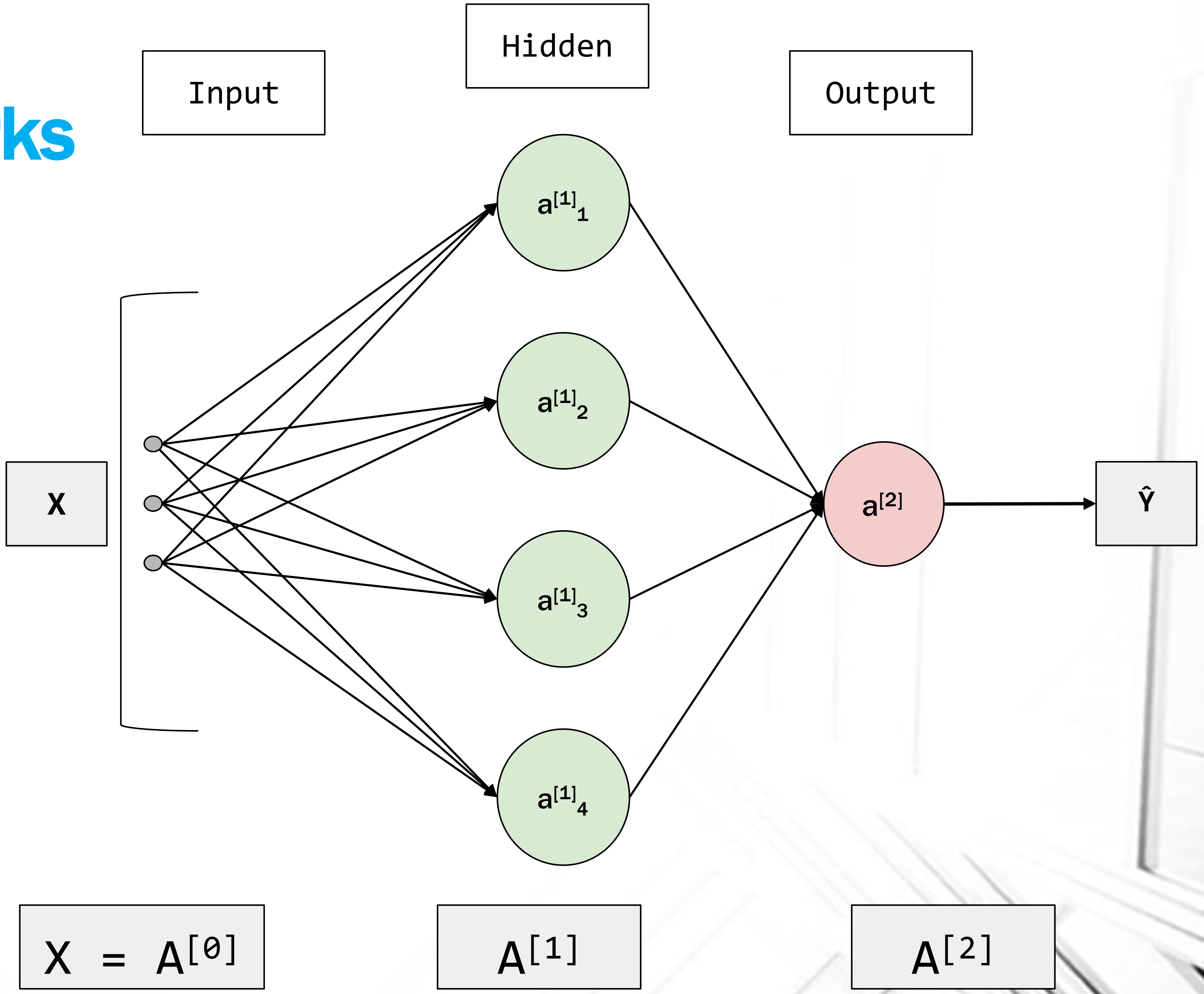
Perceptrons



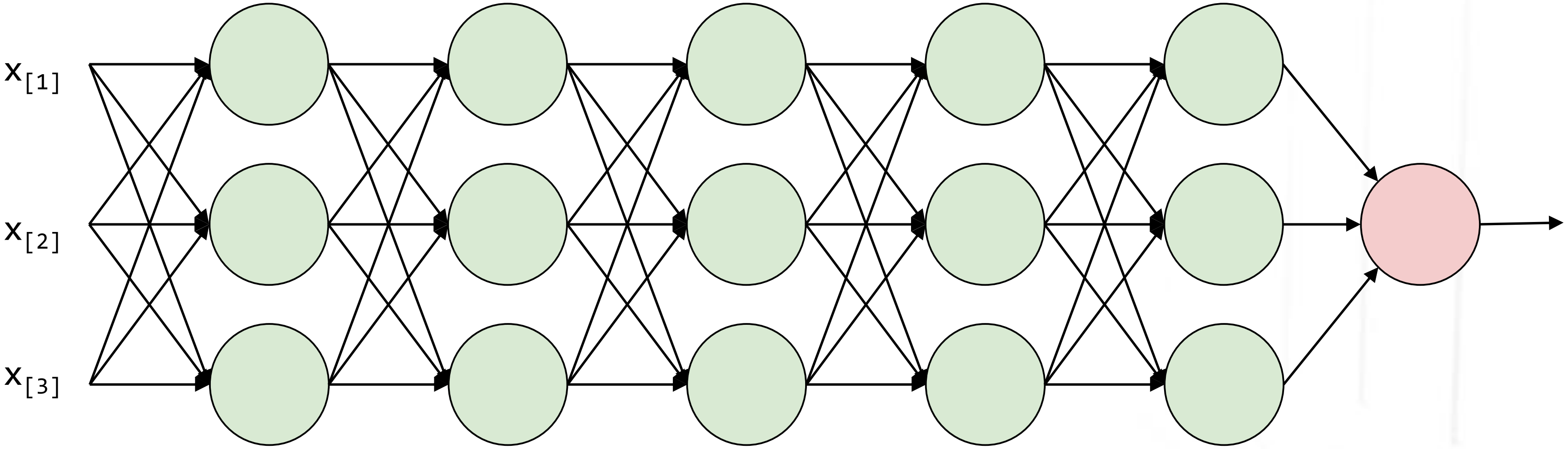


Neural Networks

One hidden layer neural network

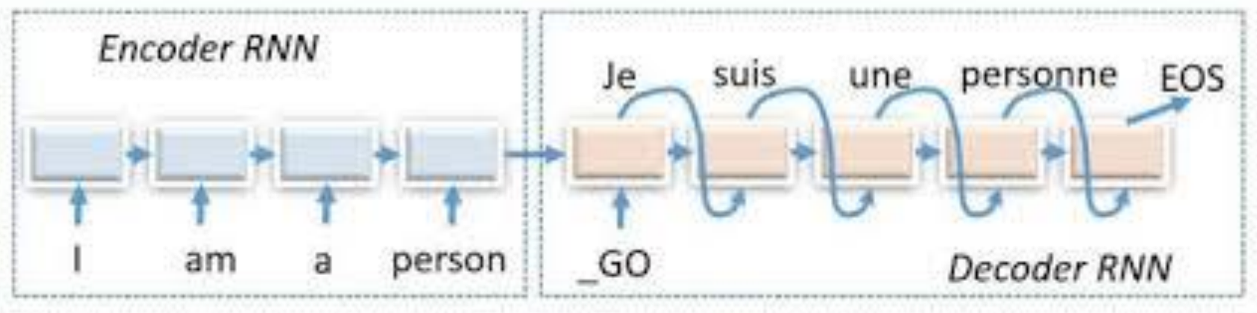


Neural Networks: going deeper



Transducing through NNs

- Networks can be used to express the intermediate states: Recurrent Neural Networks are used in this way
- States can be encoded and decoded, i.e. rewritten
- Decoding can be carried out locally (i.e. token-by-token) or globally (i.e. on a sentence-by-sentence basis)
- An Example: a transducer for Machine Translation



Encoding-Decoding

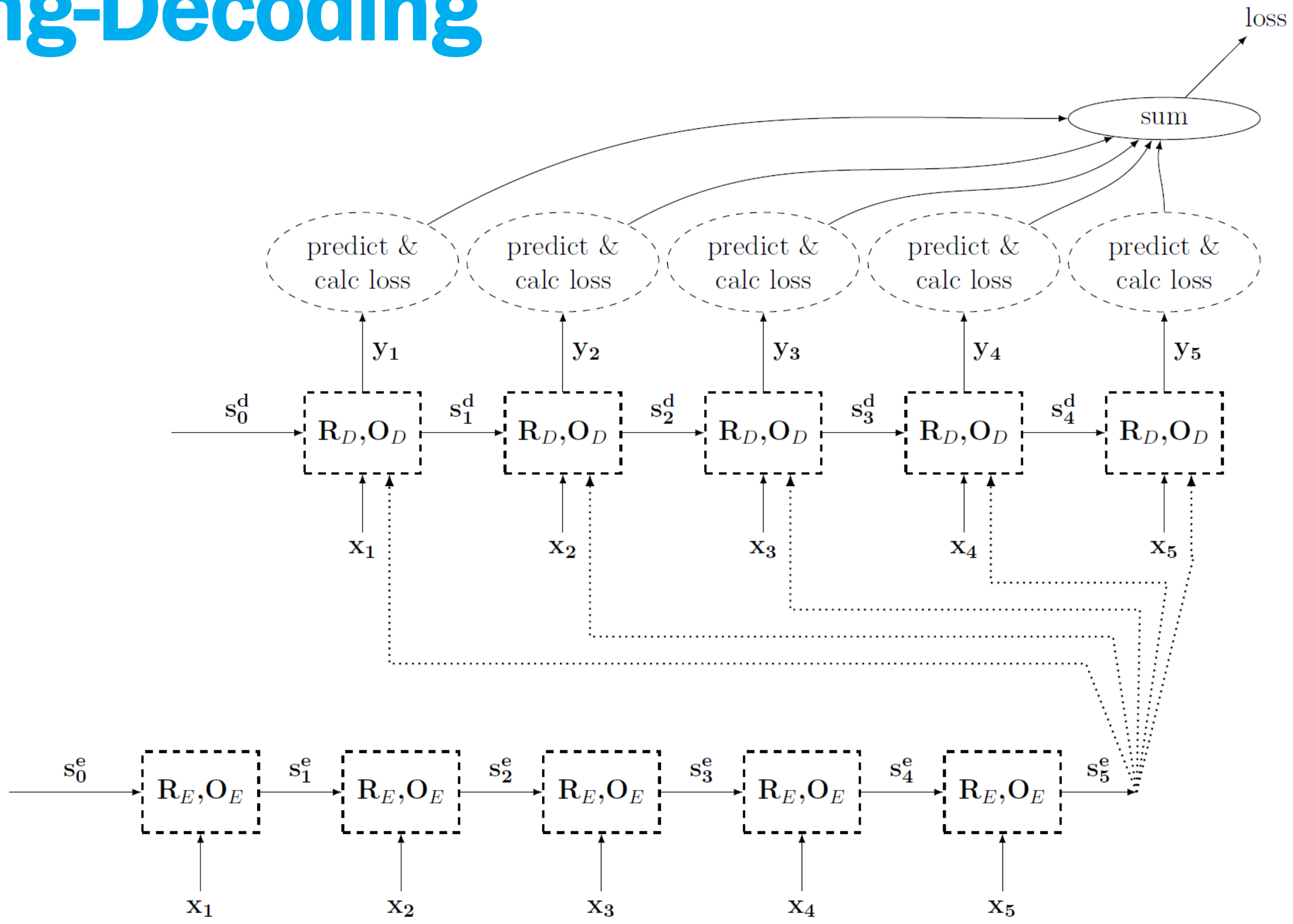


Figure 9: Encoder-Decoder RNN Training Graph.