COM4511 Speech Technology: From Points to Sequences

Anton Ragni

February 17, 2020

Why Sequences?

- From Wikipedia:

  > *A* **sequence** *is an enumerated collection of objects in which repetitions are allowed and* **order** *does matter.*

  - may have fixed or variable length
- Many (natural) phenomena have sequential nature
  - text, prices, discrete signals (speech, video)
- The nature of "objects" is important
  - discrete and continuous variables, sequences, trees, graphs

Common Problems

- Problem #1: predict next item

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{k-1} \longmapsto \boldsymbol{x}_k$$

- Problem #2: predict another sequence

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K \longmapsto \boldsymbol{y}_1, \ldots, \boldsymbol{y}_L$$

  - note that sequences may have different length
- (And related) problem #3: remove noise

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K \longmapsto \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_K$$

- (And related) problem #4: infer latent variables

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K \longmapsto q_1, \ldots, q_K$$

Name at least one task posing each sequence modelling problem

Anton Ragni

Conditional Probabilities and Chain Rule

- Conditional probabilities are critical elements of sequence modelling
  - sampling next item

$$\boldsymbol{x}_k \sim p(\boldsymbol{x}|\boldsymbol{x}_{k-1}, \ldots, \boldsymbol{x}_1)$$

  - predicting most likely next item

$$\hat{\boldsymbol{x}}_k = \arg\max_{\boldsymbol{x}} \{p(\boldsymbol{x}|\boldsymbol{x}_{k-1}, \ldots, \boldsymbol{x}_1)\}$$

- Enable to compute sequence probabilities (joint probabilities) via chain rule
  - (unconditional) joint distribution

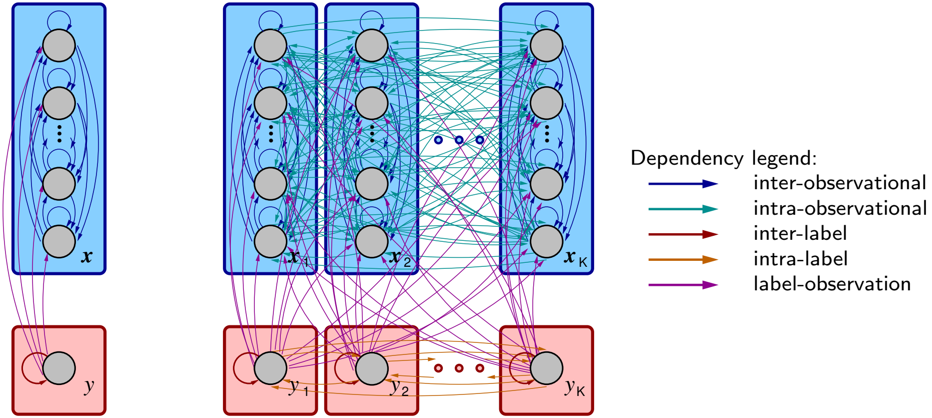
$$p(\boldsymbol{X}_{1:K}) = p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) = \prod_{k=1}^{K} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \ldots, \boldsymbol{x}_1)$$

  - conditional (joint) distribution

$$p(\boldsymbol{Y}_{1:L}|\boldsymbol{X}_{1:K}) = p(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_L|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) = \prod_{l=1}^{L} p(\boldsymbol{y}_l|\boldsymbol{y}_{l-1}, \ldots, \boldsymbol{y}_1, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_K)$$

- Name possible challenges modelling conditional probabilities

Dependency Modelling

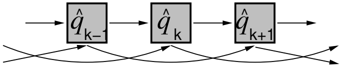


(a) Points (b) Sequences

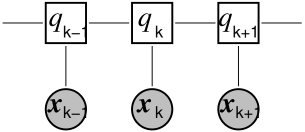
- Sequences introduce a large number of new dependencies
  - unclear which dependencies are important and how to model (form, robustness)
- Options available:
  - constrain (Markov assumption), simplify (latent variables), try to learn

(Pseudo) Dynamic Bayesian Networks (DBN)

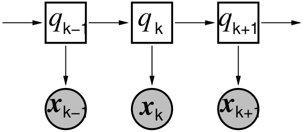
- Methodology for graphical representation of complex dependencies



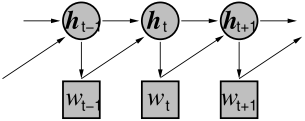
(a) DBN (trigram)



(b) DBN (CRF)



(c) DBN (HMM)



(d) Pseudo-DBN (RNN)

- Notation for DBNs:

circles	continuous variables	shaded	observed variables
squares	discrete variables	non-shaded	unobserved variables

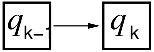
lines	general dependency
arrows	probabilistic dependency

- Pseudo DBNs use the same notation but more loose interpretation

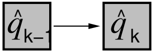
Anton Ragni

Simple DBN examples

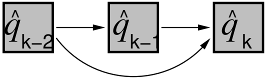
- Describe and write-down dependencies illustrated below



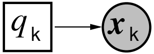
(a) ?



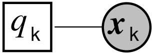
(b) ?



(c) ?



(d) ?



(e) ?

?

(f) $p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, q_k)$

- Notation for DBNs:

circles	continuous variables	shaded	observed variables
squares	discrete variables	non-shaded	unobserved variables

lines	general dependency
arrows	probabilistic dependency

Anton Ragni

Markov Assumption

- n-th order Markov assumption

$$p(\boldsymbol{X}_{1:K}) = \prod_{k=1}^{K} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\ldots,\boldsymbol{x}_1) \approx \prod_{k=1}^{K} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\ldots,\boldsymbol{x}_{k-n})$$

  - current event does not depend on events further than n steps in the past
  - which distributions can be used to model these probabilities? if $\boldsymbol{X}_{1:K}$ are discrete?
- Zeroth order Markov assumption

$$p(\boldsymbol{X}_{1:K}) \approx \prod_{k=1}^{K} p(\boldsymbol{x}_k)$$

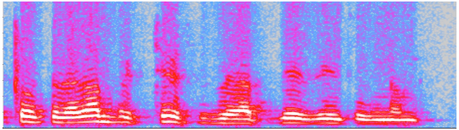
  - events are independent — too radical simplification
- First-order Markov assumption

$$p(\boldsymbol{X}_{1:K}) \approx \prod_{k=1}^{K} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$$

  - though dependencies restricted still need to know how to model — options?

Anton Ragni

Latent Variables



(a) Observed variables

`the cat sat on the mat`

(b) Word representation $\boldsymbol{z}_{1:M}$

`t h e c a ... m a t`

(c) Graphemic representation $\boldsymbol{z}_{1:L}$

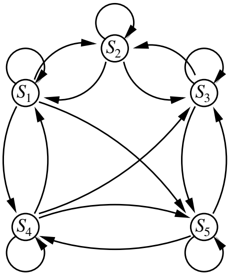
- Observed variables $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K$ often hard to interpret and explain
  - physical realisations of natural phenomena (speech!)
- The underlying process often have lower dimensional, latent, representation
  - conceptual, syntactic, word, graphemic, phonetic
- Generally link between observed and latent representations unknown
  - introduce unobserved latent variables to link two representations

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K \longrightarrow q_1, \ldots, q_K \qquad \text{(same time scale)}$$
$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K \longrightarrow q_1, \ldots, q_L \qquad \text{(different time scales)}$$

  - BUT need to know how to model these variables

Anton Ragni

Markov Process



- Key elements
  - states: $S_1, \ldots, S_N$
  - initial state distribution: $\boldsymbol{\pi} = P(S_1), \ldots, P(S_N)$
  - transition probabilities: $\boldsymbol{\Pi} = \{P(S_i|S_j)\}_{\substack{i=1\ldots N,\\ j=1\ldots N}}$
- Probability of state sequence

$$P(\boldsymbol{q}_{1:K}) \approx \prod_{k=1}^{K} P(q_k|q_{k-1})$$

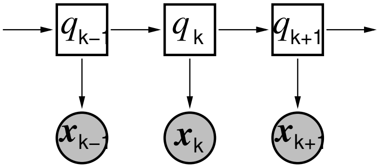
- Example #1: compute probability of state sequence S_2, S_1, S_5, S_4, S_5, S_3, S_3

$$\boldsymbol{\pi} = \begin{bmatrix} 0 \\ 1/3 \\ 2/3 \\ 0 \\ 0 \end{bmatrix}, \qquad \boldsymbol{\Pi} = \begin{bmatrix} 1/5 & 2/5 & 0 & 1/5 & 1/5 \\ 3/5 & 1/5 & 1/5 & 0 & 0 \\ 0 & 1/5 & 3/5 & 0 & 1/5 \\ 1/5 & 0 & 1/5 & 1/5 & 2/5 \\ 0 & 0 & 2/5 & 2/5 & 1/5 \end{bmatrix}$$

- Example #2: implications of full, band-diagonal, upper/lower triangular matrices

Anton Ragni

Hidden Markov Model – Next lecture will explore this model in details



Elements

- states $\boldsymbol{q}_{1:K}$: hidden discrete variables
- observations $\boldsymbol{X}_{1:K}$: discrete/continuous variables
- dependencies: probabilistic

- Joint probability distribution of observed and hidden sequences

$$p(\boldsymbol{X}_{1:K}, \boldsymbol{q}_{1:K}) = p(\boldsymbol{X}_{1:K}|\boldsymbol{q}_{1:K})P(\boldsymbol{q}_{1:K}) \approx \prod_{k=1}^{K} p(\boldsymbol{x}_k|q_k)P(q_k|q_{k-1})$$

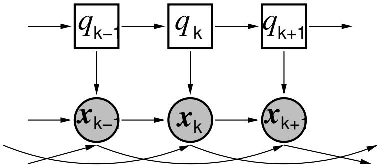
  - states are independent given past states — limits possible dependencies
  - observations are independent given current states — limits possible dependencies

- Multiple options for modelling state emission probabilities $p(\boldsymbol{x}_k|q_k)$
  - Gaussian mixture models, neural networks — how?

What are the biggest limitations of HMMs?

Anton Ragni

Auto-Regressive HMM



Elements

- states $\boldsymbol{q}_{1:K}$: hidden discrete variables
- observations $\boldsymbol{X}_{1:K}$: discrete/continuous variables
- dependencies: probabilistic

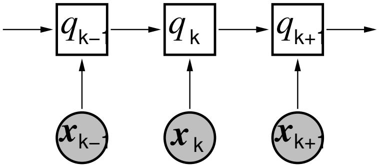
- Modified form of joint probability distribution

$$p(\boldsymbol{X}_{1:K}, \boldsymbol{q}_{1:K}) \approx \prod_{k=1}^{K} p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \ldots, \boldsymbol{x}_{k-n+1}, q_k) P(q_k | q_{k-1})$$

  - relaxes conditional independence assumptions of observations BUT not states
- State emission probabilities condition on fixed window of past observations
  - need simple and efficient form to model
- Discuss possible forms of state emission probabilities and associated issues

Anton Ragni

Maximum Entropy Markov Model



Elements

- states $\boldsymbol{q}_{1:K}$: observed discrete variables
- observations $\boldsymbol{X}_{1:K}$: discrete/continuous variables
- dependencies: probabilistic

- Conditional (!) probability distribution of latent variables given observed variables

$$P(\boldsymbol{q}_{1:K}|\boldsymbol{X}_{1:K}) = \prod_{k=1}^{K} P(q_k|\boldsymbol{q}_{1:k-1}, \boldsymbol{X}_{1:K}) \approx \prod_{k=1}^{K} P(q_k|q_{k-1}, \boldsymbol{x}_k)$$

  - states are independent given past states and current observations
- Maximum entropy model yields distribution over next states

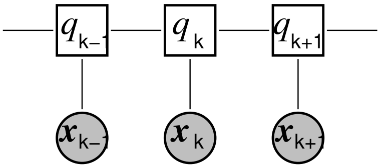
$$P(q_k|q_{k-1}, \boldsymbol{x}_k) \triangleq \exp\left(\boldsymbol{\alpha}^{\mathsf{T}}\phi(q_k, q_{k-1}, \boldsymbol{x}_k)\right)/Z(q_{k-1}, \boldsymbol{x}_k)$$

  - effective approach for combining diverse features (discrete q_k and continuous $\boldsymbol{x}_k$)
  - BUT states with low number of transitions effectively ignore observations (label bias)

Discuss which features can be used by maximum entropy models?

Anton Ragni

(Linear Chain) Conditional Random Fields



Elements

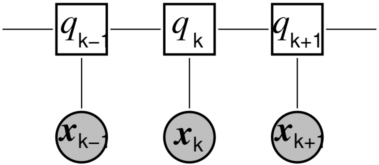
- states $\boldsymbol{q}_{1:K}$: observed discrete variables
- observations $\boldsymbol{X}_{1:K}$: discrete/continuous variables
- dependencies: general

- Alternative form for conditional probability

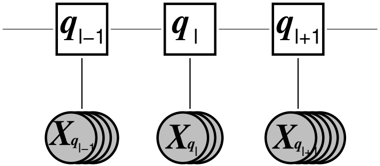
$$P(\boldsymbol{q}_{1:K}|\boldsymbol{X}_{1:K}) \triangleq \frac{1}{Z(\boldsymbol{X}_{1:K})} \prod_{k=1}^{K} \exp(\boldsymbol{\alpha}^{\mathsf{T}} \phi(q_k, q_{k-1}, \boldsymbol{x}_k))$$

  - employ the same assumption as MEMMs but do not need to be locally normalised
  - normalisation term $Z(\boldsymbol{x}_{1:K})$ can be efficiently computed — why?
- Still makes use of crude conditional independence assumptions
  - Discuss possible options to overcome these assumptions

Semi-Markov Process



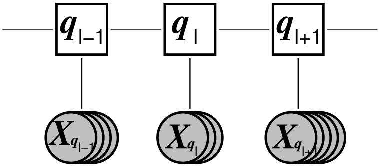
(a) A Markov process



(a) A Semi-Markov process

- Markov assumptions though highly efficient are very crude
  - limit the range of possible dependencies
  - impact the choice of latent variables
- Semi-Markov models enable to relax these assumptions
  - latent variables: Markovian dependencies
  - observed variables: non-Markovian dependencies
- Need to decide
  - what are latent variables?
  - how to link them with observed variables? (segmentation)
  - how to infer/marginalise segmentation?

Segmental Conditional Random Fields/Conditional Augmented Models



Elements:

- states $\boldsymbol{Q}_{1:L}$: hidden discrete variables
- observations $\boldsymbol{X}_{1:K}$: discrete/continuous variables
- dependencies: general

- Partition observed sequence into L segments

$$\boldsymbol{X}_{1:K}\big|_{\boldsymbol{Q}_{1:L}} = \underbrace{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{|\boldsymbol{q}_1|}}_{\boldsymbol{q}_1}, \ldots, \underbrace{\boldsymbol{x}_{|\boldsymbol{Q}_{1:l-1}|+1}, \ldots, \boldsymbol{x}_{|\boldsymbol{Q}_{1:l}|}}_{\boldsymbol{q}_l}, \ldots, \underbrace{\boldsymbol{x}_{|\boldsymbol{Q}_{1:L-1}|+1}, \ldots, \boldsymbol{x}_K}_{\boldsymbol{q}_L}$$

  - discuss available options

- Use CRF-style formulation to yield probability of segmentation

$$P(\boldsymbol{q}_{1:L}|\boldsymbol{X}_{1:K}) = \frac{1}{Z(\boldsymbol{X}_{1:K})} \prod_{l=1}^{L} \exp(\boldsymbol{\alpha}^{\mathsf{T}} \boldsymbol{\phi}(q_{l-1}, q_l, \boldsymbol{X}_{\boldsymbol{q}_l}))$$

  - though dependencies within $\boldsymbol{X}_{\boldsymbol{q}_l}$ restricted, need to know how to extract — options?

Anton Ragni

History Modelling (Dependencies Revisited)

- Previous approaches attempt to constrain and simplify possible dependencies
  - independence assumptions known to be false for many problems (including speech!)
  - modelling even simple dependencies challenging
- Alternatively, encode all dependencies into a compact representation

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{k-1} \longrightarrow \boldsymbol{h}_{k-1}$$

  - need to handle variable length sequences and model long-span dependencies
  - discuss possible issues
- Yields simple form of conditional probabilities

$$p(\boldsymbol{X}_{1:K}) = \prod_{k=1}^{K} p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \ldots, \boldsymbol{x}_1) \approx \prod_{k=1}^{K} p(\boldsymbol{x}_k | \boldsymbol{h}_{k-1})$$

- Multiple possibilities for learning optimal representations
  - recursion (recurrent neural network), attention (encoder-decoder neural network)

SPandH

Anton Ragni

Recursion

- General form of recursion to map $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{k-1}$ to $\boldsymbol{h}_{k-1}$

$$\boldsymbol{h}_{k-1} = \phi(\boldsymbol{x}_{k-1}, \boldsymbol{h}_{k-2})$$

  - $\boldsymbol{h}_{k-1}$ is a function of all past observations!
  - discuss options for $\boldsymbol{h}_0$ and ϕ
- Examples
  - simple recurrent unit

$$\boldsymbol{h}_{k-1} = \phi(\boldsymbol{W}\boldsymbol{x}_{k-1} + \boldsymbol{V}\boldsymbol{h}_{k-2} + \boldsymbol{b})$$

  - gated recurrent unit

$$\boldsymbol{h}_{k-1} = \boldsymbol{u}_{k-1} \odot \boldsymbol{h}_{k-2} + (\boldsymbol{1} - \boldsymbol{u}_{k-1}) \odot \phi(\boldsymbol{W}\boldsymbol{x}_{k-1} + \boldsymbol{V}(\boldsymbol{r}_{k-1} \odot \boldsymbol{h}_{k-2}) + \boldsymbol{b})$$

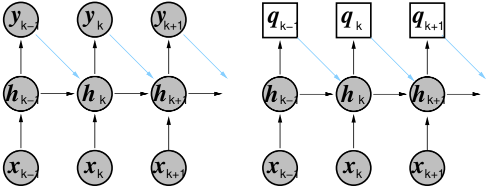
    - update gate (reset gate $\boldsymbol{r}_{k-1}$ has similar form)

$$\boldsymbol{u}_{k-1} = \boldsymbol{\sigma}(\boldsymbol{W}^{(u)}\boldsymbol{x}_{k-1} + \boldsymbol{V}^{(u)}\boldsymbol{h}_{k-2} + \boldsymbol{b}^{(u)})$$

  - numerous other forms have been examined

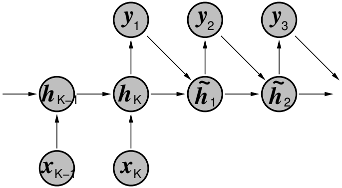
Anton Ragni

Sequence Prediction



(a) $p(\boldsymbol{Y}_{1:K}|\boldsymbol{X}_{1:K})$ (b) $P(\boldsymbol{q}_{1:K}|\boldsymbol{X}_{1:K})$

(a) Instantaneous



(b) Delayed $p(\boldsymbol{Y}_{1:L}|\boldsymbol{X}_{1:K})$

- Use recursive history representation to simplify sequence prediction

$$p(\boldsymbol{Y}_{1:K}|\boldsymbol{X}_{1:K}) = \prod_{k=1}^{K} p(\boldsymbol{y}_k|\boldsymbol{Y}_{1:k-1},\boldsymbol{X}_{1:K}) \approx \prod_{k=1}^{K} p(\boldsymbol{y}_k|\boldsymbol{Y}_{1:k-1},\boldsymbol{X}_{1:k}) \approx \prod_{k=1}^{K} p(\boldsymbol{y}_k|\boldsymbol{h}_k)$$

  - time-synchronous prediction
- Alternatively, delay prediction till all observed variables have been seen

$$p(\boldsymbol{Y}_{1:L}|\boldsymbol{X}_{1:K}) = \prod_{k=1}^{L} p(\boldsymbol{y}_l|\boldsymbol{Y}_{1:l-1},\boldsymbol{X}_{1:K}) \approx \prod_{k=1}^{L} p(\boldsymbol{y}_l|\boldsymbol{Y}_{1:l-1},\boldsymbol{h}_K) \approx \prod_{k=1}^{L} p(\boldsymbol{y}_l|\widetilde{\boldsymbol{h}}_{l-1})$$

  - suitable for sequences with different time scales ($K \neq L$)

Anton Ragni

Handling Latent Variables

- Latent sequences $\boldsymbol{q}_{1:L}$ provide one of many possible links between $\boldsymbol{X}_{1:K}$ and $\boldsymbol{z}_{1:M}$
  - though interesting, probabilities $p(\boldsymbol{X}_{1:K}, \boldsymbol{q}_{1:L})$ and $P(\boldsymbol{q}_{1:L}|\boldsymbol{X}_{1:K})$, are of limited use
- Options available for dealing with latent variables
  - (a) marginalise over all sequences, (b) find most likely sequence
- Each model classes requires different treatment
  - generative models (HMM,AR-HMM)

$$p(\boldsymbol{X}_{1:K}|\boldsymbol{z}_{1:M}) = \bigoplus_{\boldsymbol{q}_{1:K} \in \boldsymbol{Q}_{1:K}} p(\boldsymbol{X}_{1:K}, \boldsymbol{q}_{1:K}|\boldsymbol{z}_{1:M}) = \bigoplus_{\boldsymbol{q}_{1:K} \in \boldsymbol{Q}_{1:K}} p(\boldsymbol{X}_{1:K}|\boldsymbol{q}_{1:K}, \boldsymbol{z}_{1:M}) P(\boldsymbol{q}_{1:K}|\boldsymbol{z}_{1:M})$$

    - note that previously omitted conditioning on $\boldsymbol{z}_{1:M}$ is made explicit here
  - discriminative models (MEMM,CRF,SCRF/CAug)

$$P(\boldsymbol{z}_{1:M}|\boldsymbol{X}_{1:K}) = \bigoplus_{\boldsymbol{q}_{1:L} \in \boldsymbol{Q}_{1:L}} P(\boldsymbol{z}_{1:M}, \boldsymbol{q}_{1:L}|\boldsymbol{X}_{1:K}) \approx \bigoplus_{\boldsymbol{q}_{1:L} \in \boldsymbol{Q}_{1:L}} P(\boldsymbol{z}_{1:M}|\boldsymbol{q}_{1:L}) P(\boldsymbol{q}_{1:L}|\boldsymbol{X}_{1:K})$$

    - discriminative alignment model $P(\boldsymbol{z}_{1:M}|\boldsymbol{q}_{1:L})$
- Next lecture will examine latent variables in HMMs

Anton Ragni

Summary

- Sequence data
  - highly ubiquitous (text, speech, market prices)
  - significant increase in the number and type of possible dependencies
  - cannot be handled using standard distance-based and probabilistic models
- Sequence models
  - attempt to constrain, simplify or model possible dependencies
  - Markov assumptions are key methodology for constraining the scope of dependencies
  - latent variables are key for interpreting complex types of observed variables
  - recursive, learnable, history representations provide an interesting alternative