Spoken language understanding: Semantic decoding

Milica Gašić

Dialogue Systems and Machine Learning Group,
Heinrich Heine University Düsseldorf

In this lecture...

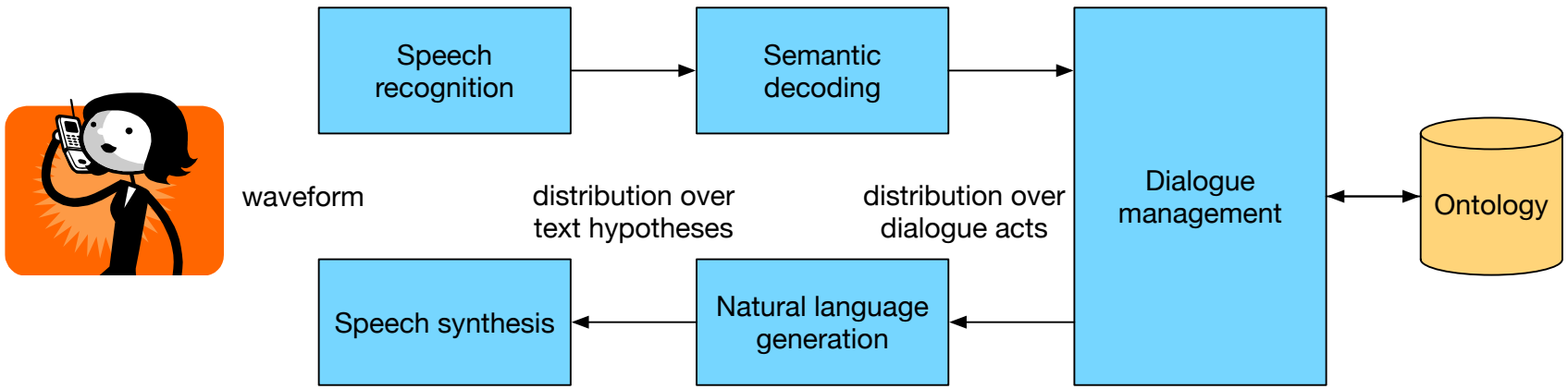
Dialogue acts

Semantic decoding as a classification task

Input features to semantic decoder

Semantic decoding as a sequence to sequence learning task

Architecture of a statistical spoken dialogue system



Problem

Decoding meaning in utterances:

- *Do they serve Korean food*
- *Can you repeat that please*
- *Hi I want to find a restaurant that serves Italian food*
- *How about a restaurant that serves Lebanese food*
- *I want a different restaurant*
- *Is it near Union Square*
- *May I have the address*
- *No, I want an expensive restaurant*

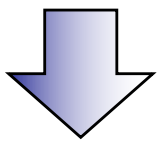
Reminder: Dialogue acts

Semantic concepts:

dialogue act type - encodes the system or the user intention in a (part of) dialogue turn

semantic slots and values - further describe entities from the ontology that a dialogue turn refers to

Is there um maybe a cheap place in the centre of town please?



inform (price = cheap, area = centre)

dialogue act type

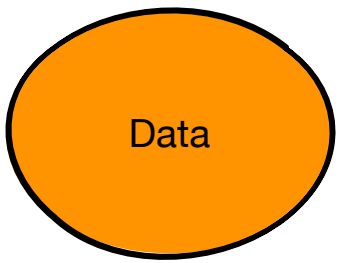
semantics slots and values

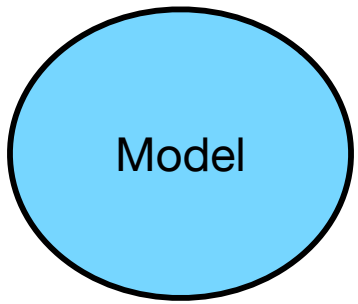
Semantic decoding

- *Do they serve Korean food*
- *Can you repeat that please*
- *Hi I want to find an Italian restaurant*
- *I want a different restaurant*
- *Is it near Union Square*
- *May I have the address*
- *No, I want an expensive restaurant*
- *How about a restaurant that serves Lebanese food*

- confirm(food=Korean)
- repeat()
- hello(type=restaurant, food=Italian)
- reqalts()
- confirm(near=Union Square)
- request(addr)
- negate(type=restaurant, pricerange=expensive)
- reqalts(type=restaurant, food=Lebanise)

Semantic decoding

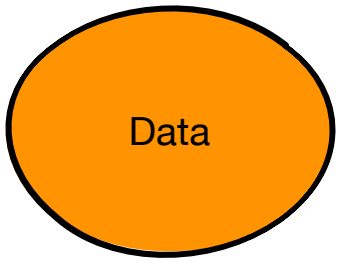




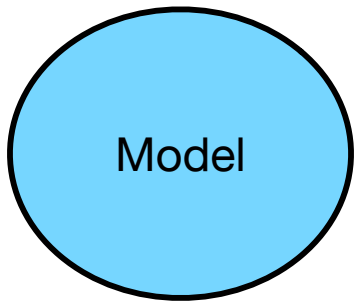
- Dialogue utterances labelled with semantic concepts

- The set of semantic concepts

Semantic decoding as a classification task



- Dialogue utterances labelled with semantic concepts

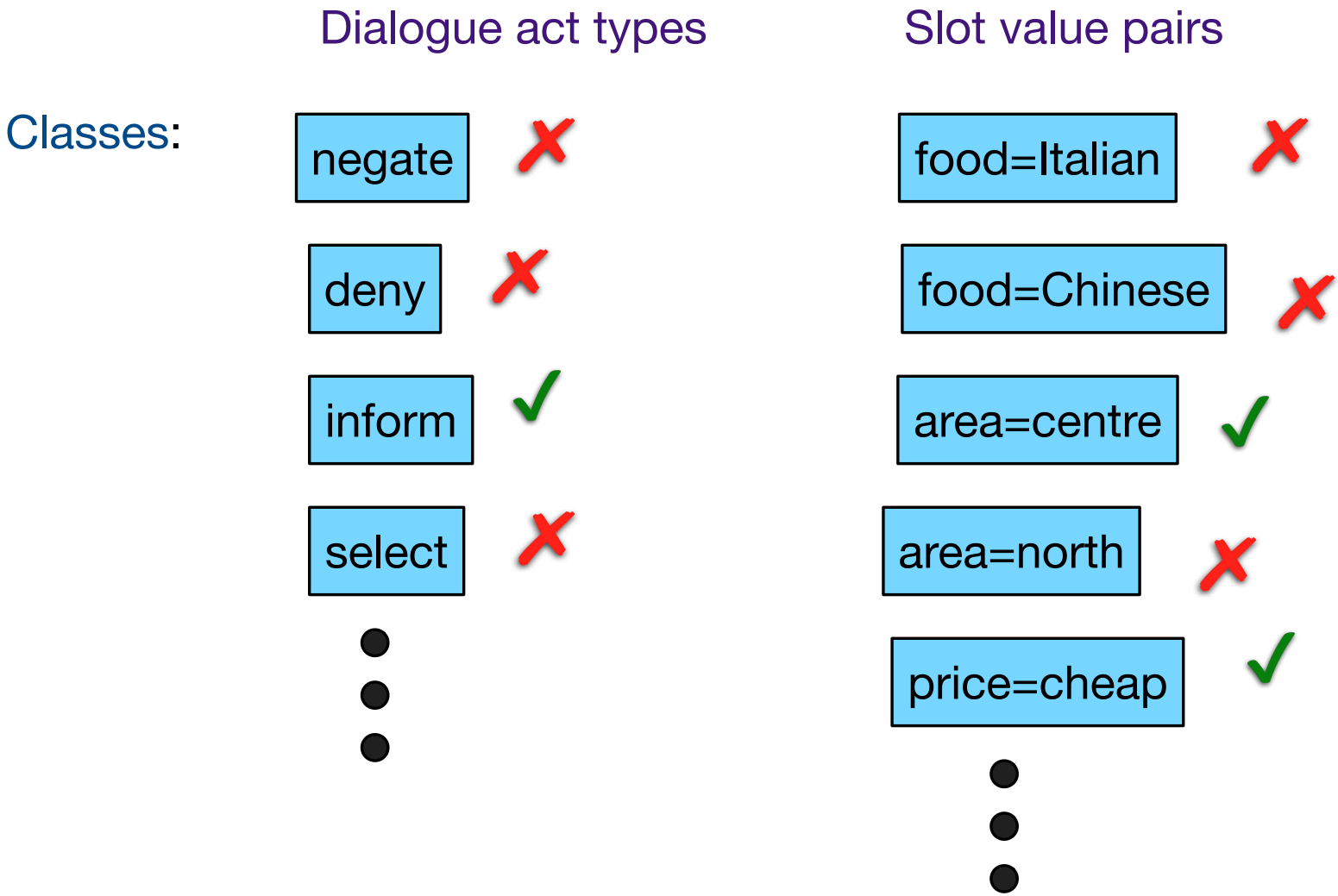


- Support vector machines

- The set of semantic concepts

Semantic decoding as a classification task

Is there um maybe a cheap place in the centre of town please?



Theory: support vector machines

- Support vector machine is a maximum margin classifier
- Support vectors are input data points that lie on the margin
- Input data points are mapped into a high dimensional feature space where the data is linearly separable.

$$\mathbf{x} \rightarrow \phi(\mathbf{x})$$

- Kernel function is the dot product of feature functions

$$k(\mathbf{x}, \mathbf{x}) = \phi(\mathbf{x})^{\mathsf{T}} \phi(\mathbf{x})$$

Theory: support vector machines

- The decision surface is given by

$$f(\mathbf{x}) = \sum_{i=1}^{n} y_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) + \beta$$

$\mathbf{x}$	test data point
$\mathbf{x}_i$	support vectors
y_i	labels, $y_i \in \{1, -1\}$
α_i	weight of the support vector in the feature space
β	bias
$k(\cdot, \cdot)$	kernel function

Theory: support vector machines

- Extended to multiclass SVM using *one-versus-rest* approach
- The output of an SVM is transformed into probability by fitting a sigmoid

$$p(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(af(\mathbf{x}) + b)}$$

and estimating a, b by maximum likelihood on a validation set

Input to semantic decoder

top ASR hypothesis Features are extracted directly from top hypothesis and the classification is performed into relevant semantic classes.

Ontology and Delexicalisation

name(Carluccios)
food(Italian)
pricerange(moderate)
area(centre)

name(Seven Days)
food(Chinese)
pricerange(cheap)
area(centre)

name(Cocum)
food(Indian)
pricerange(cheap)
area(north)

I'm looking for an Italian restaurant.

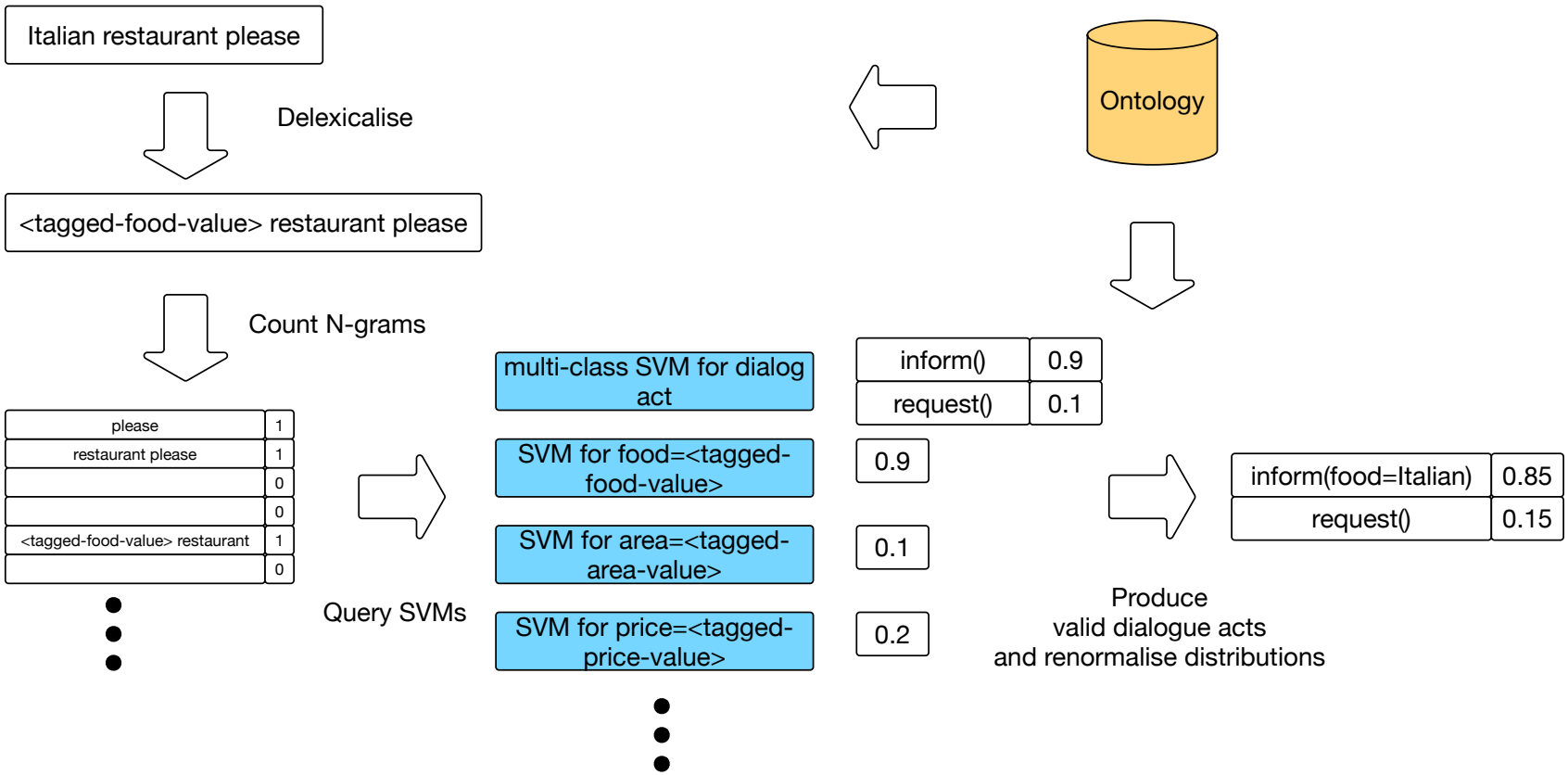
I'm looking for an Indian restaurant.

I'm looking for a Chinese restaurant.

I'm looking for an <tagged-food-value> restaurant.

I'm looking for a <tagged-food-value> restaurant.

SVMs in semantic decoding: semantic tuple classifier



Input to semantic decoder

N-best list of ASR hypothesis In real conversational systems error rate of the top hypothesis is typically 20-30%. To achieve robustness alternative hypotheses are needed.

Taking alternative ASR hypotheses into account

Is there an expensive restaurant?	0.35
Is there an inexpensive restaurant?	0.30
Inexpensive restaurant?	0.20
In expensive restaurant?	0.05

Semantic Decoding

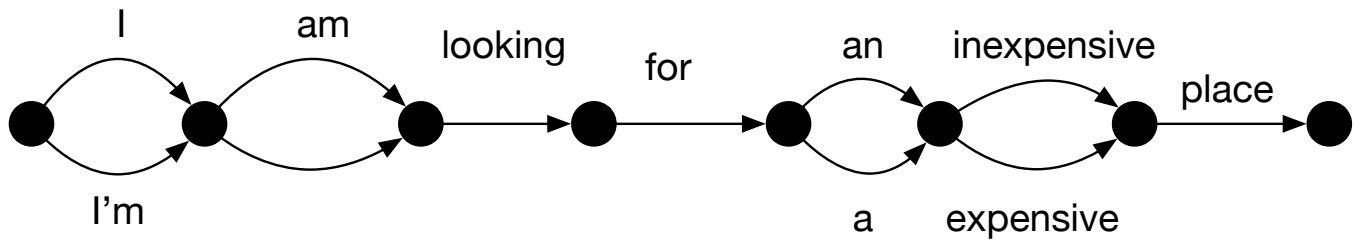
inform(type=restaurant, price=expensive)	0.35
inform(type=restaurant, price=inexpensive)	0.30
inform(type=restaurant, price=inexpensive)	0.20
inform(type=restaurant, price=expensive)	0.05

Combine outputs

inform(type=restaurant, price=inexpensive)	0.50
inform(type=restaurant, price=expensive)	0.40

Input to semantic decoder

Word confusion network summarises the posterior distribution of ASR better, without pruning low probability words. Each arc in the word confusion network has a posterior probability for that word. That is the sum of all paths which contain that word at around that approximate time.



Context features can be extracted from the last system action. The user response may be dependent on the system question. Caution! We still want to understand utterances where the user is not following the system.

Evaluate the quality of semantic decoder

F-score C_{ref} semantic concepts in reference and C_{hyp_0} semantic concepts in top hypothesis

$$F = 2\frac{|C_{ref} \cap C_{hyp_0}|}{|C_{ref}| + |C_{hyp_0}|}$$

Item level cross entropy (ICE) [Thomson et al., 2008] measures the quality of the output distribution p_i for every concept

$$ICE = \frac{1}{1 + |C_{ref}|} \sum_{c \in C} \log(p(c)p^*(c) + (1-p(c))(1-p^*(c))),$$

where $p(c) = \sum_i p_i(c), c \in C_{hyp_i}$ and

$$p^*(c) = \begin{cases} 1 & c \in C_{ref} \\ 0 & \text{otherwise} \end{cases}$$

Results [Henderson et al., 2012]

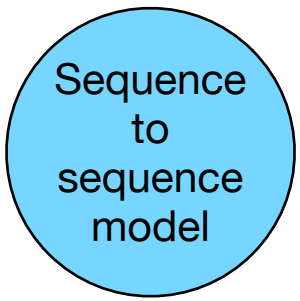
- Cambridge Restaurant Information Domain
- Semantic concepts area, price-range, food-type, phone number, post-code, signature dish, address of restaurant and dialogue act types: inform, request, confirm etc
- Data collected in car WER 37.0%

Input	F-Score	ICE
Top ASR hypothesis	0.692 ± 0.012	1.790 ± 0.065
N-best ASR hypotheses	0.708 ± 0.012	1.760 ± 0.074
Confusion network	0.730 ± 0.011	1.680 ± 0.063
Confusion network + context	0.767 ± 0.011	0.880 ± 0.063

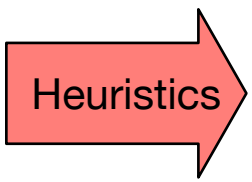
Semantic decoding as a sequence to sequence learning task

- Reads the input word by word, or window of words
- Outputs sequence of concepts using BIO labelling (begin, inside, other)
- These are then heuristically mapped into slot-value pairs

Is there um maybe a cheap place in the centre of town serving Chinese food please?

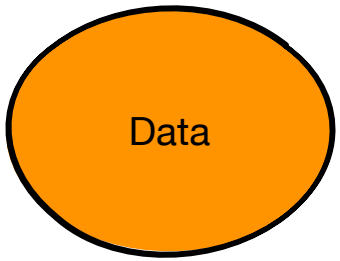


o o o o o b_price o o o b_area i_area i_area b_food i_food i_food o

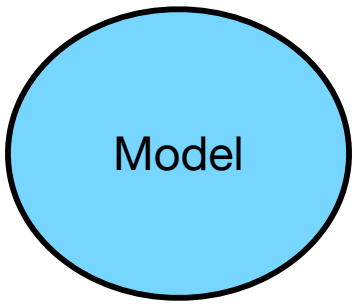


price=cheap area=centre food=Chinese

Semantic decoding as a sequence to sequence learning task



- Dialogue utterances with semantic concepts



- Conditional random fields

- The sequence of semantic concepts

Theory: Conditional random fields

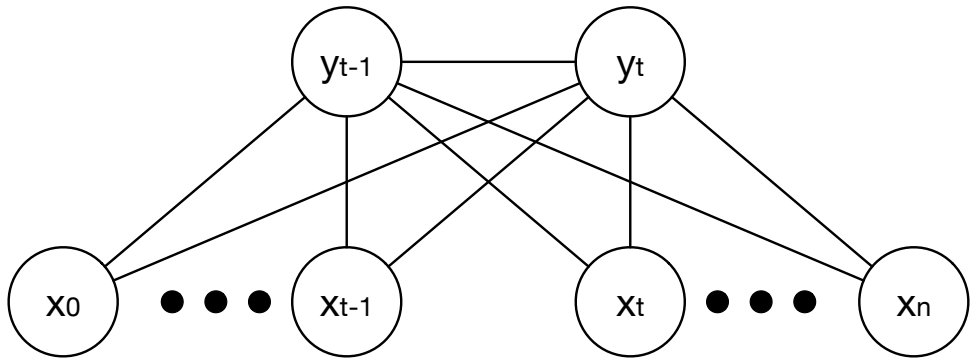
$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_i \lambda_i f_i(\mathbf{x}, \mathbf{y})\right)$$

This is a fully connected undirected graph where:

- $\mathbf{x}$ input sequence $(x_0, \cdots, x_n)$
- $\mathbf{y}$ output sequence $(y_0, \cdots, y_n)$
- f_i given feature functions
- λ_i parameters to be estimated

Theory: Linear chain conditional random field

In this case the graph is not fully connected any more, the label at time step t depends on the label in the previous time step $t-1$.



$$p(\mathbf{y}|\mathbf{x}) = \prod_t \frac{1}{Z(\mathbf{x})} \exp\left(\sum_i \lambda_i f_i(\mathbf{x}, y_t, y_{t-1}, t)\right) \tag{1}$$

$$= \frac{1}{Z(\mathbf{x})} \exp(\lambda^\mathsf{T} \mathbf{F}(\mathbf{x}, \mathbf{y})) \tag{2}$$

$$Z(\mathbf{x}) = \sum_{\mathbf{y}'} \exp(\lambda^\mathsf{T} \mathbf{F}(\mathbf{x}, \mathbf{y}')) \tag{3}$$

Training a linear chain conditional random field

- Maximise the log probability $\log p(\mathbf{y}|\mathbf{x})$ with respect to parameters λ.
- It can be shown that the gradient of the log probability is the difference between the feature function values and the expected feature function values:

$$\nabla_\lambda \mathcal{L} = \mathbf{F}(\mathbf{x}, \mathbf{y}) - \sum_{\mathbf{y}'} p(\mathbf{y}'|\mathbf{x}) \mathbf{F}(\mathbf{x}, \mathbf{y}').$$

- Since the label at each time step only depends on the label in the previous time step, message passing can be used to find the expectation.

Gradient

$$\log p(\mathbf{y}|\mathbf{x}) = \lambda^\top \mathbf{F}(\mathbf{x}, \mathbf{y}) - \log Z(\mathbf{x})$$

$$\begin{aligned}\frac{\partial}{\partial \lambda_i} \log p(\mathbf{y}|\mathbf{x}) &= \mathbf{F_i}(\mathbf{x}, \mathbf{y}) - \frac{1}{Z(\mathbf{x})} \sum_{\mathbf{y}'} \exp(\lambda^\top \mathbf{F}(\mathbf{x}, \mathbf{y}'))\mathbf{F_i}(\mathbf{x}, \mathbf{y}') \\ &= \mathbf{F_i}(\mathbf{x}, \mathbf{y}) - \sum_{\mathbf{y}'} p(\mathbf{y}'|\mathbf{x})\mathbf{F_i}(\mathbf{x}, \mathbf{y}')\end{aligned}$$

Expectation

Forward:

$$\alpha_t = \begin{cases} \alpha_{t-1} M_t & 1 \leq t \leq n \\ \mathbf{1} & t = 0 \end{cases}$$

Backward:

$$\beta_t^{\mathsf{T}} = \begin{cases} M_{t+1} \beta_{t+1}^{\mathsf{T}} & 1 \leq t \leq n \\ \mathbf{1} & t = 0 \end{cases}$$

where

$$M_t[y, y'] = \exp(\sum_i \lambda_i f_i(\mathbf{x}, y, y', t))$$

Expectation is then:

$$\sum_{\mathbf{y}'} p(\mathbf{y}'|\mathbf{x}) \mathbf{F_i}(\mathbf{y}', \mathbf{x}) = \sum_t \frac{\alpha_{t-1}(f_{ti} * M_t)\beta_t^{\mathsf{T}}}{Z(\mathbf{x})}$$

$$Z(\mathbf{x}) = \mathbf{1}^{\mathsf{T}} \alpha_n, f_{ti}[y, y'] = f_i(\mathbf{x}, y, y', t), * \text{ element-wise}$$

Linear chain CRFs in semantic decoding [Tur et al., 2013]

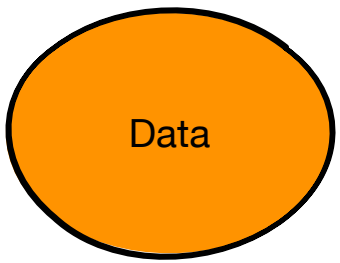
Input data Word confusion networks where each bin is annotated with semantic concept

Features For each bin in the confusion network extract N-grams of the neighbouring bins and weight them by their confidence scores.

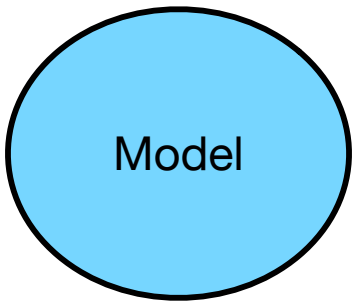
Task is conversational understanding system with real users about movies (22 concepts)

Input	F-Score
Top ASR hypothesis	0.77
Confusion network	0.83

Semantic decoding as a sequence to sequence learning task



- Dialogue utterances with semantic concepts



- Recurrent neural networks

- The sequence of semantic concepts

Theory: Neural networks

Neural network transforms input vector $\mathbf{x}$ into an output categorical probability distribution $\mathbf{y}$:

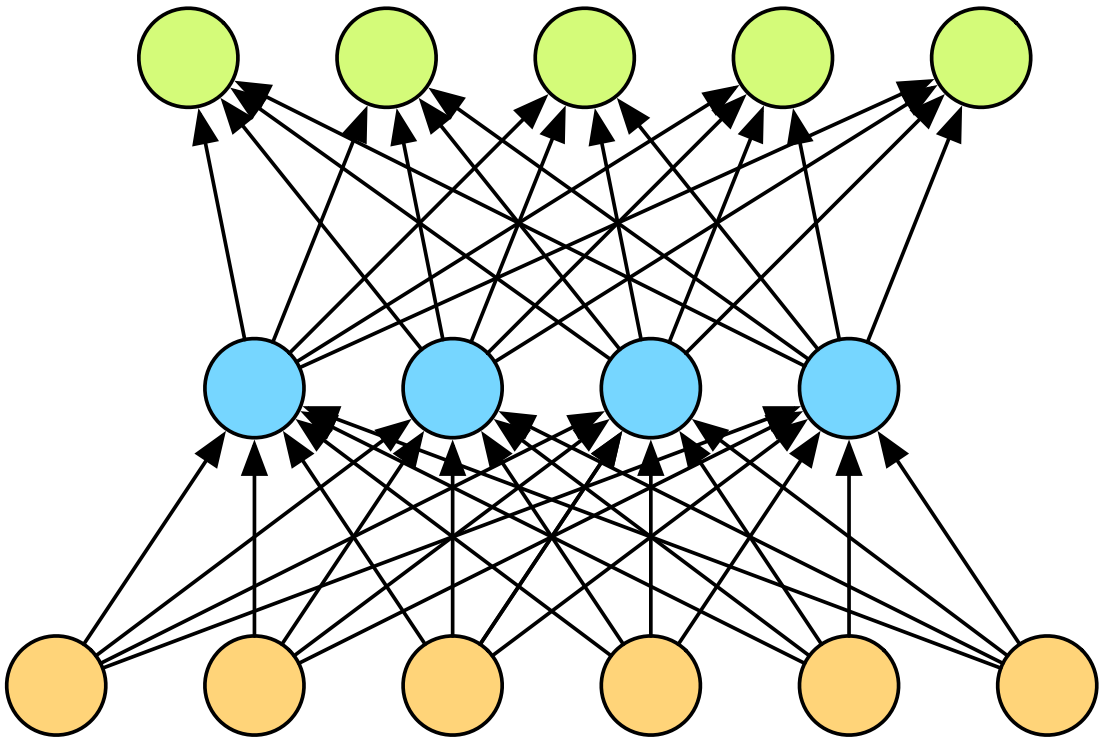
$$\begin{aligned}
\mathbf{h}_0 &= g_0(W_0\mathbf{x}^\mathsf{T} + b_0) \\
\mathbf{h}_i &= g_i(W_i\mathbf{h}_{i-1}^\mathsf{T} + b_i), 0 < i < m \\
\mathbf{y} &= \text{softmax}(W_m\mathbf{h}_{m-1}^\mathsf{T} + b_m) \\
\text{softmax}(\mathbf{h})_i &= \exp(h_i)/(\sum_j \exp(h_j))
\end{aligned}$$

where

- g_i (differentiable) activation functions hyperbolic tangent tanh or sigmoid σ
- W_i, b_i parameters to be estimated

Theory: Neural networks

Neural network structure



Theory: Training neural networks

Cost function is the negative log probability of true label
$-\sum_j y_{ij} \log y'_{ij}$

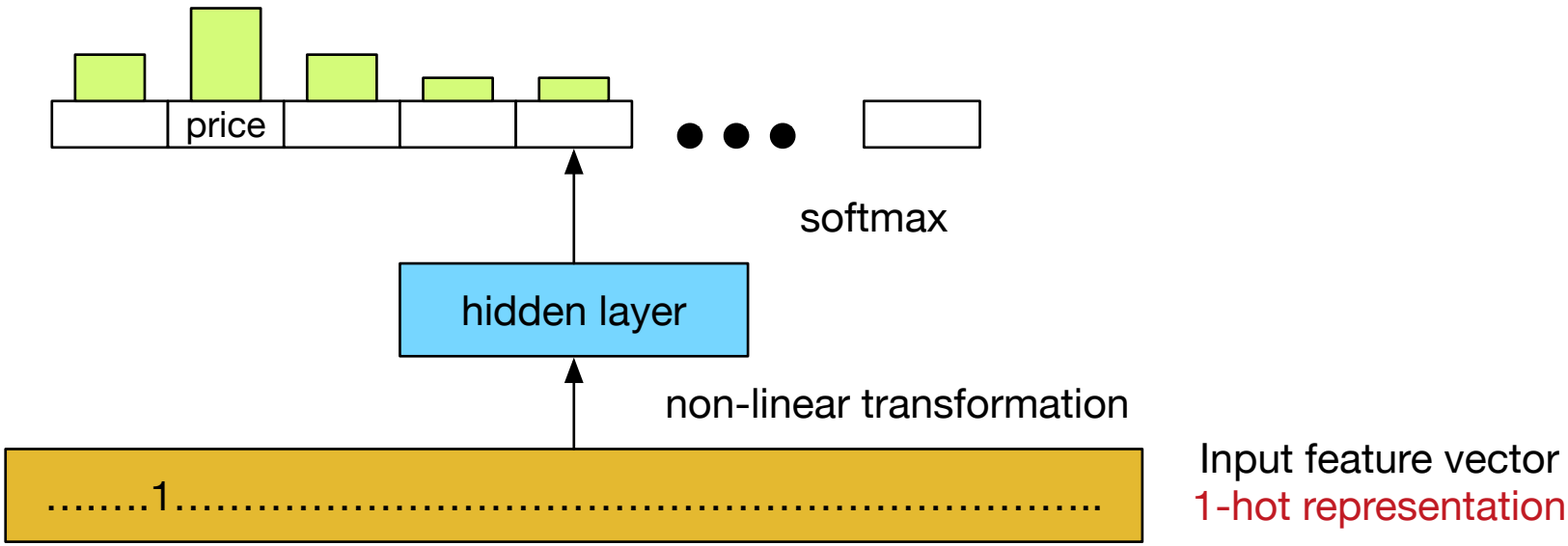
$\mathbf{y}_i$ is delta distribution (zero everywhere except for the correct category)

$\mathbf{y}'_i$ is the probability distribution estimated by a Neural network

The cost function can be minimised by stochastic gradient descent.

Neural networks for semantic decoding

Example network which does not take into account context.

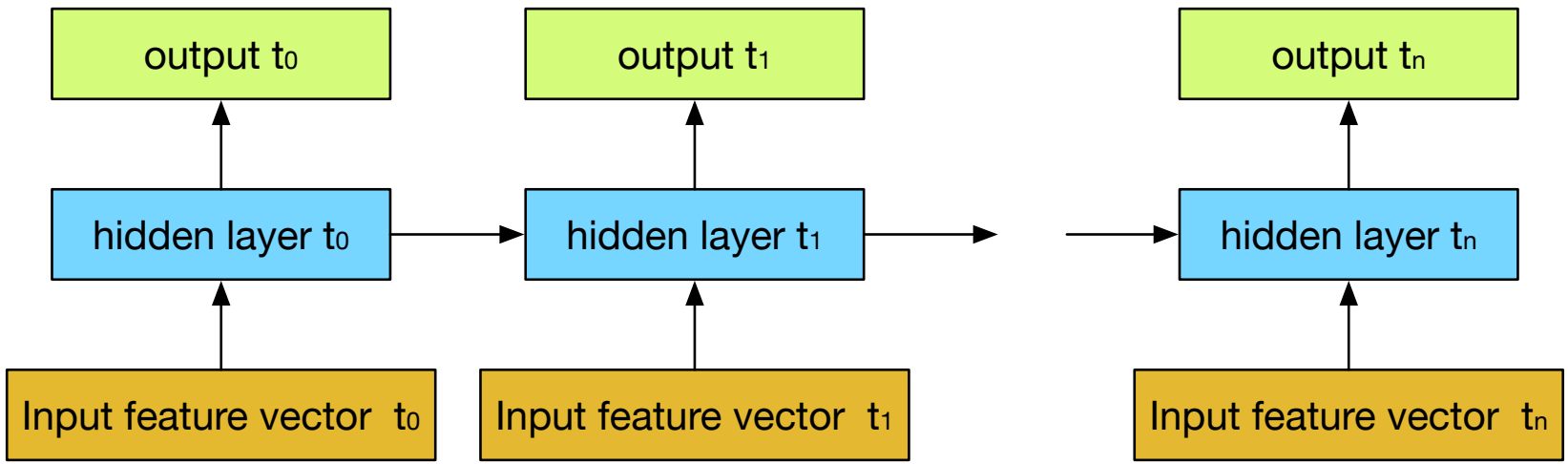


I'm looking for a <tagged-price-value> restaurant

I'm looking for a cheap restaurant

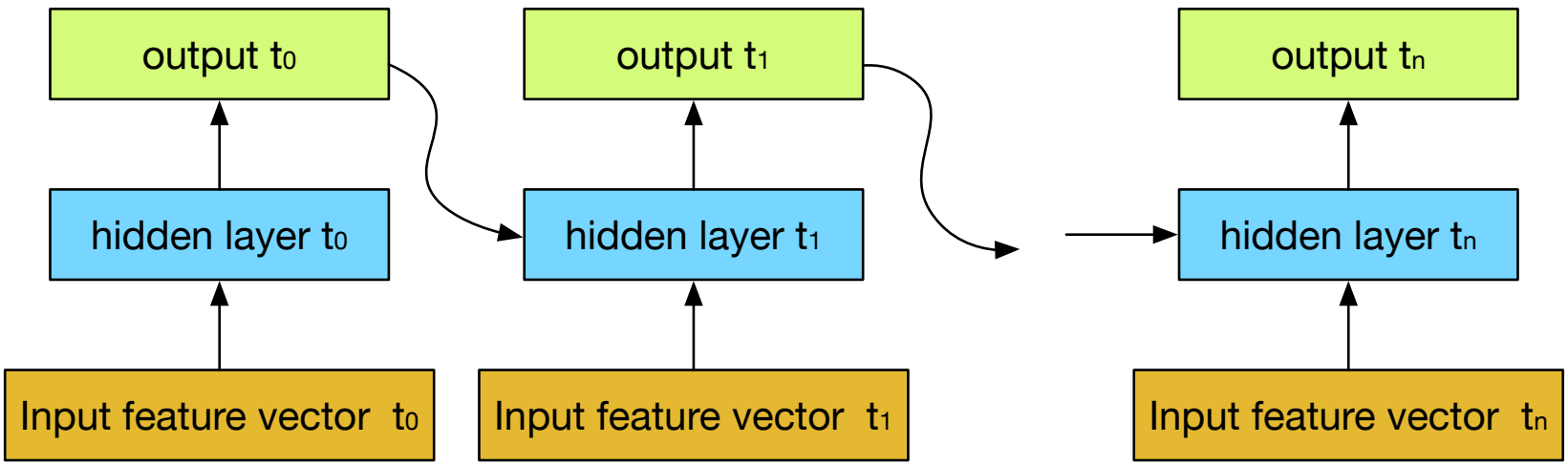
Recurrent neural networks – Elman-type

Recurrent neural networks are deep neural networks unrolled through time. Elman-type neural network has recurrent connections between hidden layers of the neural networks:



Recurrent neural networks – Jordan-type

Jordan-type neural network feeds the output of previous time step into the next time step:



RNNs in semantic decoding [Mesnil et al., 2015]

ATIS dataset - flight booking information.

- I want to fly from Boston to New York.

Input features 1-hot representation or context window.

F-score	Elman	Jordan	CRF
1-hot	0.932	0.652	0.67
window	0.950	0.942	0.929

F-score on entertainment dataset

CRF	RNN
0.906	0.881

Long short-term memory neural networks

RNNs automatically learn context information but suffer from vanishing gradient problem. Long short-term memory neural networks are an alternative model which to some extent avoid this problem and have been successfully used in semantic decoding [Yao et al., 2014].

Summary

Input

- Input can be 1-best or N-best list from the ASR or a confusion network.
- Taking into account alternative recognition result improves robustness.

Model

- Semantic decoding can be defined as a classification task.
- In this case a collection of SVMs can be used.
- Semantic decoding can be more naturally defined as a sequence to sequence learning task.
- CRFs are one sequence-to-sequence model which require predefined context feature functions.
- RNNs automatically provide context but suffer from vanishing gradient problem.

References I

Henderson, M., Gasic, M., Thomson, B., Tsiakoulis, P., Yu, K., and Young, S. (2012).
Discriminative spoken language understanding using word confusion networks.
In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 176–181.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., and Zweig, G. (2015).
Using recurrent neural networks for slot filling in spoken language understanding.
Trans. Audio, Speech and Lang. Proc., 23(3):530–539.

References II

Thomson, B., Yu, K., Gasic, M., Keizer, S., Mairesse, F., Schatzmann, J., and Young, S. (2008).
Evaluating semantic-level confidence scores with multiple hypotheses.
In *INTERSPEECH*, pages 1153–1156.

Tur, G., Deoras, A., and Hakkani-Tur, D. (2013).
Semantic parsing using word confusion networks with conditional random fields.
Annual Conference of the International Speech Communication Association (Interspeech).

References III

Yao, K., Peng, B., Zhang, Y., Yu, D., Zweig, G., and Shi, Y. (2014).
Spoken language understanding using long short-term memory neural networks.
In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 189–194.