Universität Hamburg

DER FORSCHUNG | DER LEHRE | DER BILDUNG

Dialog State Tracking using Recurrent Neural Networks

Department of Informatics

Speech Technology SS 16

Thomas Hummel

22th June 2016

Questions which will be answered

1. What is **Dialog State Tracking**?

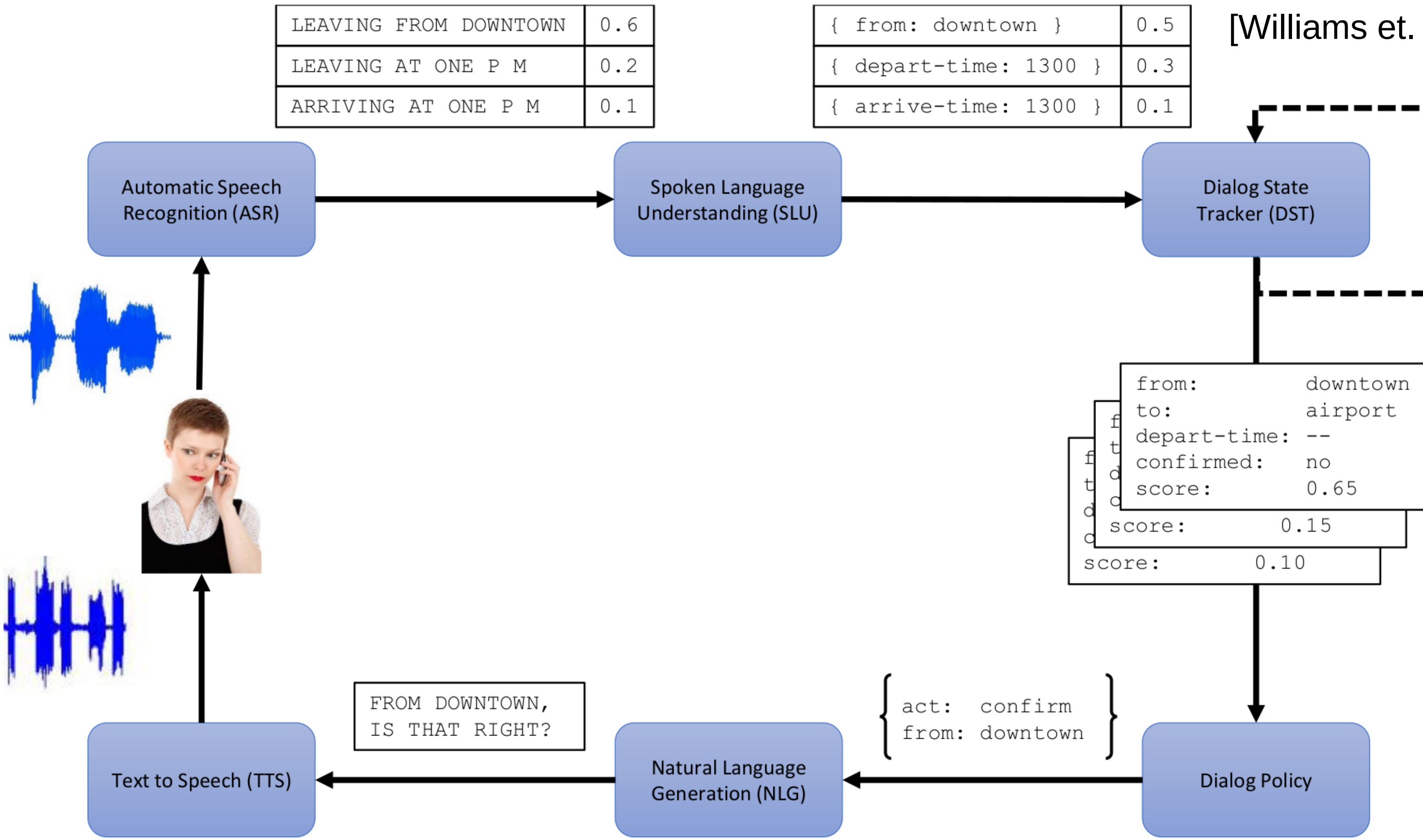
2. Which **methods** are available?

3. How can we apply Recurrent Neural Networks (**RNNs**) to DST?

4. What is the current **progress**?

Dialog State Tracking – Overview

[Williams et. al (2016)]

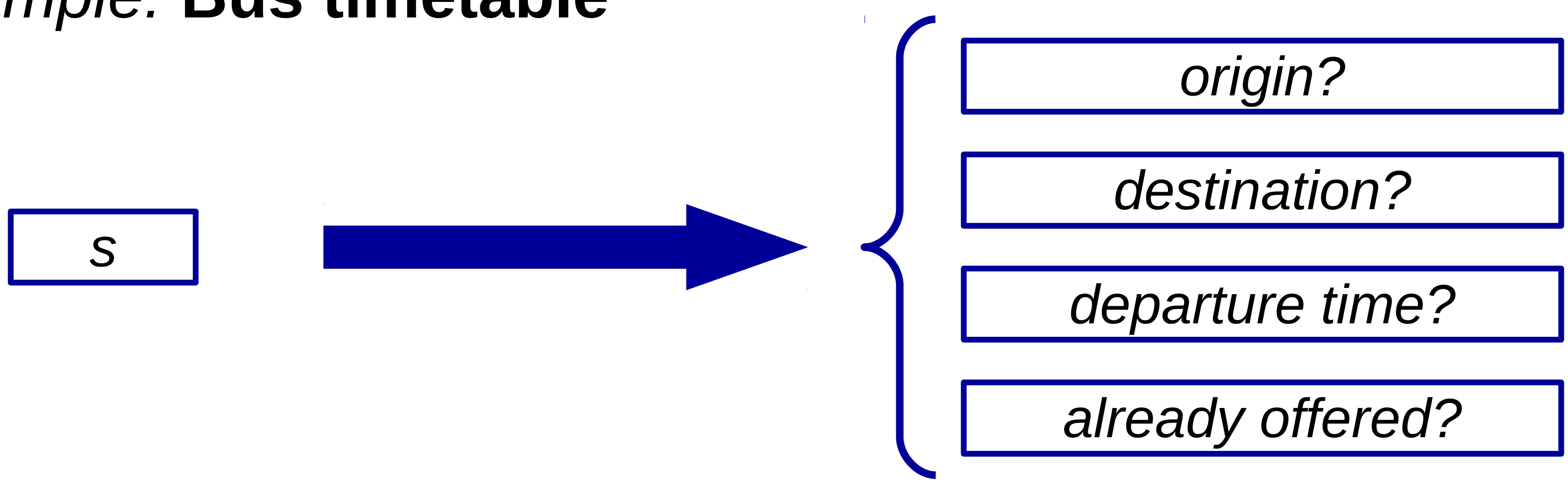


Problem definition – Dialog State

*“A **dialog state** s_t is a representation of **what the user wants** at point t **from the dialog system**”*

- encoding of **user's goal** and **relevant history**

Example: **Bus timetable**



Problem definition – State Tracking

observable elements

- ASR output
- SLU output
- …

system action history

external knowledge sources

- databases (e.g. but timetables)
- models of past dialogs
- …

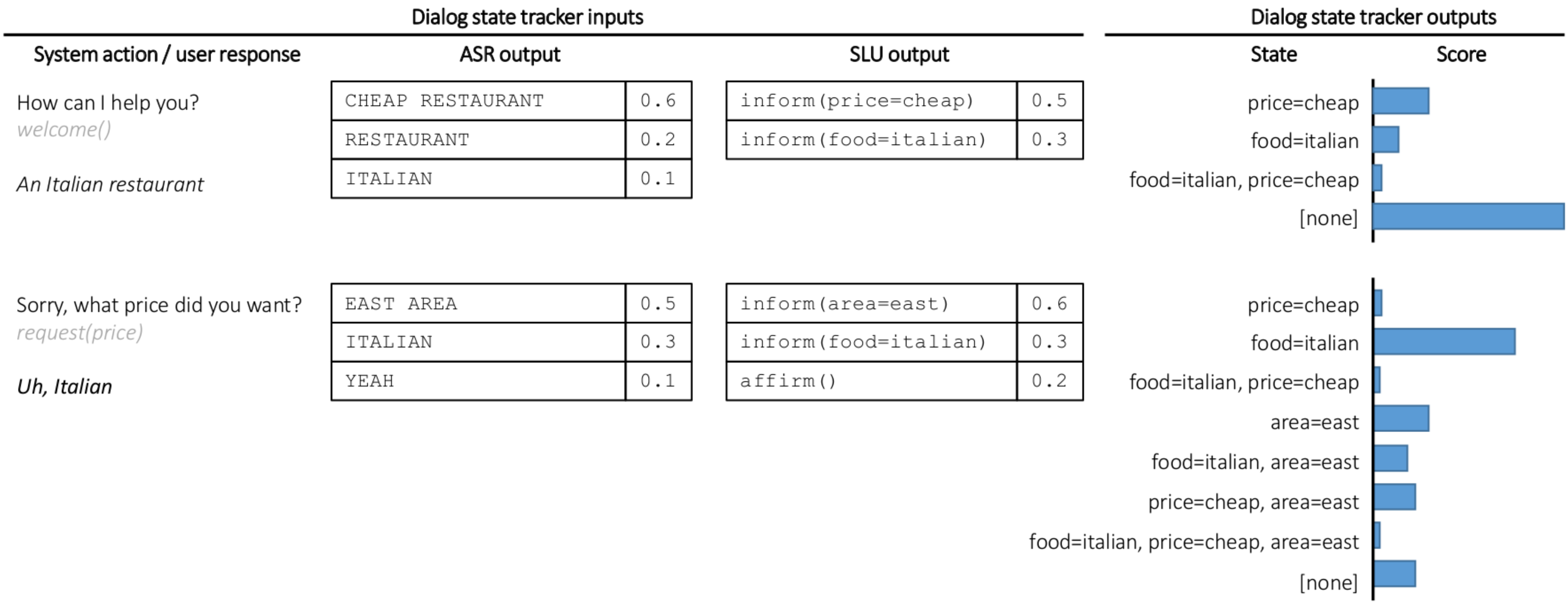
Dialog State Tracker

Current state estimation

***belief state** b(s)*

(distribution)

Problem definition – Example



[Williams et. al (2016)]

Dialog State Tracking Challenges

	Type	Domain	Innovation
DSTC 1	human-computer	bus timetable	–
DSTC 2	human-computer	restaurant information	allow goal changes
DSTC 3	human-computer	tourist information	adapt to unknown domains
DSTC 4	human-human	tourist information	increased complexity by human-human dialog

Dialog state tracking – Methods

Rule-based	*hand-crafted rules*		
Generative	*Bayesian networks, HMMs*		
Discriminative		**Static classifiers**	*SVM, Log. regression*
		Sequence models	*RNN, CRF*
		System combination	*combine multiple models*

Generative models

Goal: determine P(Y|X)

- learns a model of the **joint probability** p(X,Y)
- … to produce new samples of X and Y

Implicitly **model how the data was generated** to categorize a signal:

“Based on my generation assumption, which future state is most likely to generate the current input?”

Discriminative models

Goal: determine P(Y|X)

- learns the **conditional probability** distribution p(Y|X)
- assume only a model of how Y depends on X, not on X

Discriminative models do not care about how the data was generated, it **simply categorizes a given signal**.

Generative vs. Discriminative

Example: *Classify a speech to a language*

1) Determine the difference in the linguistic models without learning the languages and then classify the speech

Discriminative

2) Learn each language and then classify it using the knowledge just gained

Generative

Discuss!

Which method (discriminative or generative) is probably more suited for DST?

Comparing DST methods

Rule-based methods

- \+ require no data to implement
- – do not account uncertainty
- – rules require domain expertise

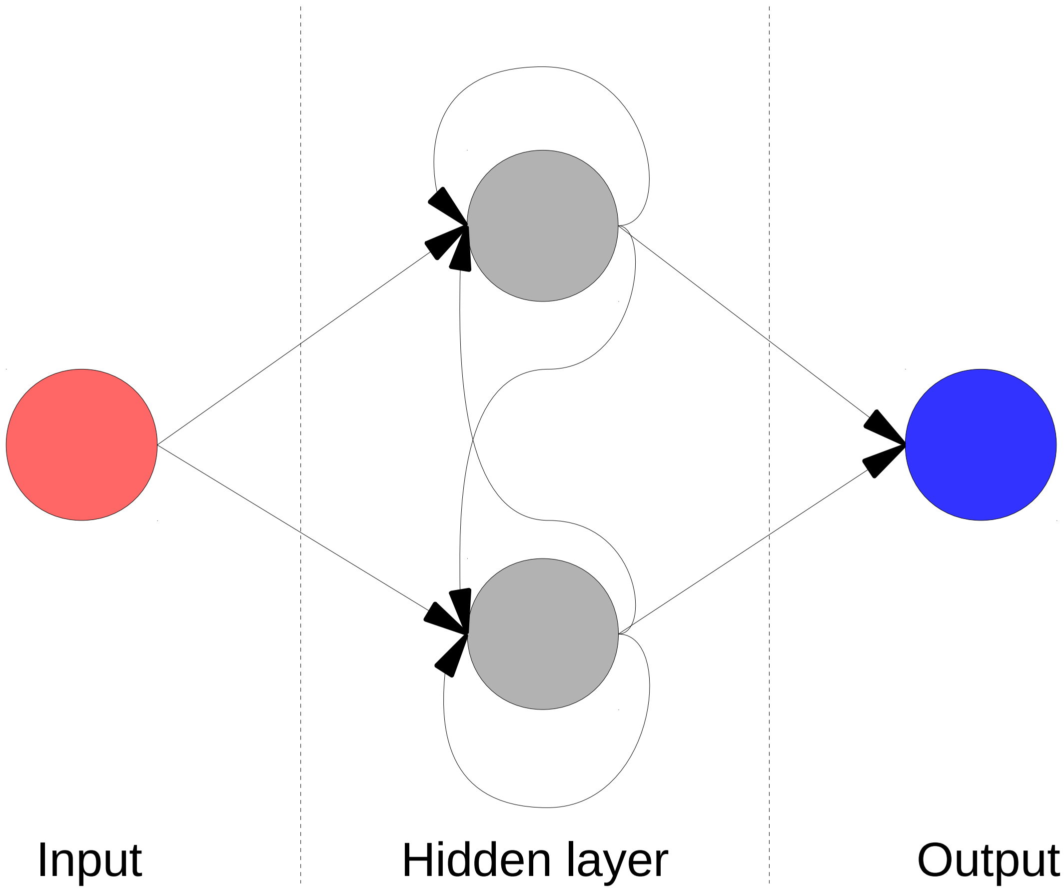
Generative methods

- \+ theoretically learns the model behind the data
- – enumerating all possible dialog states computationally not feasible
- – invalid independence assumptions

Discriminative methods

- \+ trained directly on the data and explicitly optimized for prediction accuracy
- \+ can incorporate large number of features
- \+ possible to work directly on ASR output
- – does not necessarily learn the model behind the data

Recurrent Neural Networks



Word-based DST with RNNs (1)

- RNNs provide a **natural model** for state tracking in dialog
- *word-based* state tracking by omitting SLU [Henderson et. al (2014)]

Feature representation (input)

- extracting **n-grams from utterances and dialog acts**
- for each hypothesis:
  - calculate N-best list, unigram, bigram and trigram features
  - weight n-grams by N-best probabilities and sum to vector
- vector as the input to the RNN (high-dimensional features)

Word-based DST with RNNs (2)

How to deal with states, which have not been seen in training?
(e.g. accurately recognise any possible food type)

Generalisation to unseen states

- Embed a network which **learns a generic model** of the updated belief
- *particularly*: learn a function of 'tagged' features
  (e.g. **learn the hypothesis food='***<value>***'** for any food-type replacing '<*value*>')

➡ each network consists of **two sub-networks**
I. general behaviour of tagged hypotheses
II. corrections due to correlations with untagged features

Word-based DST with RNNs (3)

Networks output: estimation of the current dialog state

Multiple runs of RNN training give results with high variability

- ***score averaging***:
  - avarage the output of ~10 individual RNNs with varying hyperparameters (regularization, learning rates, hidden layer sizes, …)

➡ **Boosting the system performance** by exploiting variability

Evaluating Dialog-State Trackers

Popular evaluation metrics

- **Accuracy**: % of turns where top-ranked hypothesis is correct
  ("correctness")

- **L2**: distance between vector of estimates and optimal vector
  ("quality of scores")

Results of DSTC 2

	ASR features	SLU features	Accuracy	L2
Bayesian net.	–	+	0.675	0.550
Linear CRF	–	+	0.601	0.648
Word-based RNN	+	–	0.768	0.346
Web-style ranking	+	+	0.784	0.735

Conclusion

- **discriminative** machine-learned (ML) methods are now the **state-of-the art in DST**
- modelling **dialog as a sequence** is natural and advantageous
- including **ASR features** in DST improves performance
- **poor generalization** and **over-tuning** to the training data is still a key issue for ML methods

Ultimate goal of a universal spoken dialog system that can converse naturally on any subject!

Literature

[Williams et. al (2016)]
J. D. Williams, A. Raux, and M. Henderson, “The Dialog State Tracking Challenge Series: A Review,” *Dialogue & Discourse*, Apr. 2016.

[Ng and Jordan (2002)]
A. Ng and M. Jordan, “On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes,” *Adv. Neural Inf. Process. Syst.*, vol. 14, p. 841, 2002.

[Henderson et. al (2015)]
M. Henderson, “Machine Learning for Dialog State Tracking: A Review,” *Proc. First Int. Work. Mach. Learn. Spok. Lang. Process.*, 2015.

[Henderson et. al (2014)]
M. Henderson, B. Thomson, and S. J. Young, “Word-based Dialog State Tracking with Recurrent Neural Networks,” in *Proceedings of SIGdial*, 2014.

Questions?

Thank you for your attention!