Deng · Liu

Deep Learning in Natural Language Processing

Li Deng · Yang Liu

Deep Learning in Natural Language Processing

Springer

Preface

Natural language processing (NLP), which aims to enable computers to process human languages intelligently, is an important inter-discipline field crossing artificial intelligence, computing science, cognitive science, information processing, and linguistics. Concerned with interactions between computers and human languages, NLP applications such as speech recognition, dialog systems, information retrieval, question answering, and machine translation have started to reshape the way people identify, obtain, and make use of information.

The development of NLP can be described in terms of three major waves: rationalism, empiricism, and deep learning. In the first wave, rationalist approaches advocated the design of hand-crafted rules to incorporate knowledge into NLP systems based on the assumption that knowledge of language in the human mind is fixed in advance by generic inheritance. In the second wave, empirical approaches assume that rich sensory input and the observable language data in surface form are required and sufficient to enable the mind to learn the detailed structure of natural language. As a result, probabilistic models were developed to discover the regularities of languages from large corpora. In the third wave, deep learning exploits hierarchical models of non-linear processing, inspired by biological neural systems to learn intrinsic representations from language data, in ways that aim to simulate human cognitive abilities.

The intersection of deep learning and natural language processing has resulted in striking successes in practical tasks. Speech recognition is the first industrial NLP application that deep learning has strongly impacted. With the availability of large-scale training data, deep neural networks achieved dramatically lower recognition errors than the traditional empirical approaches. Another prominent successful application of deep learning in NLP is machine translation. End-to-end neural machine translation that models the mapping between human languages using neural networks has proven to improve translation quality substantially. Therefore, neural machine translation has quickly become the new de facto technology in major commercial online translation services: Google, Microsoft, Facebook, Baidu, and more. Many other areas of NLP, including language understanding and dialogue, lexical analysis and parsing, knowledge graph, information retrieval, question answering

from text, social computing, language generation, and text sentiment analysis, have also seen much significant progress using deep learning, riding on the third wave of NLP. Nowadays, deep learning is a dominating method applied to practically all NLP tasks.

The main goal of this book is to provide a comprehensive survey on the recent advances in deep learning applied to NLP. The book presents state-of-the-art of NLP-centric deep learning research, and focuses on the role of deep learning played in major NLP applications including spoken language understanding, dialogue systems, lexical analysis, parsing, knowledge graph, machine translation, question answering, sentiment analysis, social computing, and natural language generation (from images). This book is suitable for readers with a technical background in computation, including graduate students, post-doctoral researchers, educators, and industrial researchers and anyone interested in getting up to speed with the latest techniques of deep learning associated with NLP.

The book is organized into eleven chapters as follows:

- Chapter 1: A Joint Introduction to Natural Language Processing and to Deep Learning (Li Deng and Yang Liu)
- Chapter 2: Deep Learning in Conversational Language Understanding (Gokhan Tur, Asli Celikyilmaz, Xiaodong He, Dilek Hakkani-Tür, and Li Deng)
- Chapter 3: Deep Learning in Spoken and Text-based Dialogue Systems (Asli Celikyilmaz, Li Deng, and Dilek Hakkani-Tür)
- Chapter 4: Deep Learning in Lexical Analysis and Parsing (Wanxiang Che and Yue Zhang)
- Chapter 5: Deep Learning in Knowledge Graph (Zhiyuan Liu and Xianpei Han)
- Chapter 6: Deep Learning in Machine Translation (Yang Liu and Jiajun Zhang)
- Chapter 7: Deep Learning in Question Answering (Kang Liu and Yansong Feng)
- Chapter 8: Deep Learning in Sentiment Analysis (Duyu Tang and Meishan Zhang)
- Chapter 9: Deep Learning in Social Computing (Xin Zhao and Chenliang Li)
- Chapter 10: Deep Learning in Natural Language Generation from Images (Xiaodong He and Li Deng)
- Chapter 11: Epilogue (Li Deng and Yang Liu)

Chapter 1 first reviews the basics of NLP as well as the main scope of NLP covered in the following chapters of the book, and then goes in some depth into the historical development of NLP summarized as three waves and future directions. Then, an in-depth survey on the recent advances in deep learning applied to NLP is organized into nine separate chapters, each covering a largely independent application area of NLP. The main body of each chapter is written by leading researchers and experts actively working in the respective field.

The origin of this book was the set of comprehensive tutorials given at the 15th China National Conference on Computational Linguistics (CCL 2016) held in October 2016 in Yantai, Shandong, China, where both of us, editors of this book, were active participants and were taking leading roles. We thank our Springer's senior editor, Dr. Celine Lanlan Chang, who kindly invited us to create this book and who

has been providing much of timely assistance needed to complete this book. We are grateful also to Springer's Assistant Editor, Jane Li, for offering invaluable help through various stages of manuscript preparation.

We thank all authors of Chapters 2-10 who devoted their valuable time carefully preparing the content of their chapters: Gokhan Tur, Asli Celikyilmaz, Dilek Hakkani-Tur, Wanxiang Che, Yue Zhang, Xianpei Han, Zhiyuan Liu, Jiajun Zhang, Kang Liu, Yansong Feng, Duyu Tang, Meishan Zhang, Xin Zhao, Chenliang Li, and Xiaodong He. The authors of Chapters 4-9 are CCL 2016 tutorial speakers. They spent a considerable amount of time in updating their tutorial material with the latest advances in the field since October 2016.

Further, we thank numerous reviewers and readers, Sadaoki Furui, Andrew Ng, Fred Juang, Ken Church, Haifeng Wang, Hongjiang Zhang, who not only gave us much needed encouragements but also offered many constructive comments which substantially improved earlier drafts of the book.

Finally, we give our appreciations to our organizations, Microsoft Research and Citadel (for Li Deng) and Tsinghua University (for Yang Liu), who provided excellent environments, supports, and encouragements that have been instrumental for us to complete this book.

Li Deng, Seattle, USA
Yang Liu, Beijing, China
October, 2017

Chapter 3
Deep Learning in Spoken and Text-Based Dialogue Systems

Asli Celikyilmaz[1], **Li Deng**[2], **and Dilek Hakkani-Tür**[2]
[1] **Microsoft Research, Redmond, Washington, USA**
[2] **Citadel, Chicago and Seattle, USA**
[3] **Google, Mountain View, California, USA**
{**asli, l.deng, dilek**}**@ieee.org**

Abstract Last few decades have witnessed substantial breakthroughs on several areas of speech and language understanding research, specifically for building human to machine conversational dialog systems. Dialogue systems, also known as interactive conversational agents, virtual agents or sometimes chat-bots, are useful in a wide range of applications ranging from technical support services to language learning tools and entertainment. Recent success in deep neural networks has spurred the research in building data-driven dialog models. In this chapter, we present state-of-the-art neural network architectures and details on each of the components of building a successful dialog system using deep learning tools. Task-oriented dialog systems would be the focus of this chapter, and later different networks are provided for building open-ended non-task oriented dialog systems. Furthermore, to facilitate research in this area, we have a survey of publicly available datasets and software tools suitable for data-driven learning of dialogue systems. Finally, appropriate choice of evaluation metrics are discussed for the learning objective.

3.1 Introduction

In the past decade, virtual personal assistants (VPAs) or conversational chat-bots have been the most exciting technological developments. Spoken Dialog Systems (SDS) are considered the brain of these VPAs. For instance Microsoft's Cortana[1], Apple's Siri[2], Amazon Alexa[3], Google Home[4], and Facebook's M[5], have incorpo-

[1] https://www.microsoft.com/en-us/mobile/experiences/cortana/
[2] http://www.apple.com/ios/siri/
[3] https://developer.amazon.com/alexa
[4] https://madeby.google.com/home
[5] https://developers.facebook.com/blog/post/2016/04/12/bots-for-messenger/

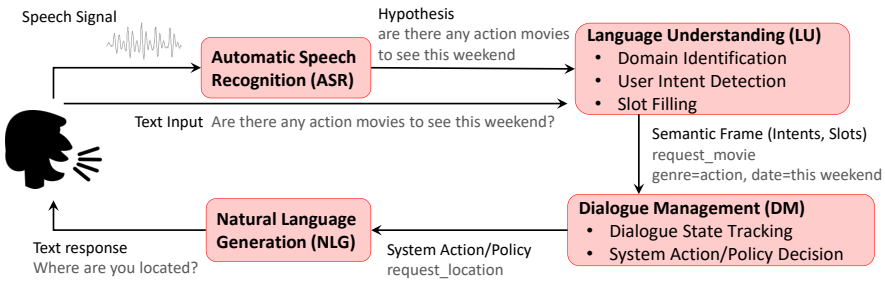

Fig. 3.1: Pipeline framework of spoken dialog system.

rated SDS modules in various devices, which allow users to speak naturally in order to finish tasks more efficiently. The traditional conversational systems have rather complex and/or modular pipelines. The advance of deep learning technologies has recently risen the applications of neural models to dialogue modeling.

Spoken dialogue systems have nearly 30 years of history, which can be divided into three generations: symbolic-rule or template based (before late 90s), statistical learning based, and deep learning based (since 2014). This chapter briefly surveys the history of conversational systems, and analyzes why and how the underlying technology moved from one generation to the next. Strengths and weaknesses of these three largely distinct types of bot technology are examined and future directions are discussed.

Current dialog systems are trying to help users on several tasks to complete daily activities, play interactive games, and even be a companion (see examples in Table 3.1). Thus, conversational dialog systems have been built for many purposes, however, a meaningful distinction can be made between goal-oriented dialogs (e.g, for personal assistant systems or other task completion dialogs such as purchasing or technical support services) and non-goal oriented dialog systems such as chit-chat, computer game characters (avatars), etc. Since they serve for different purses, structurally their dialog system designs and the components they operate on are different. In this chapter, we will provide details on the components of dialog systems for task (goal) oriented dialog tasks. Details of the non-goal oriented dialog systems (chit-chat) will also be provided.

As shown in Figure 3.1, the classic spoken dialog systems incorporate several components including Automatic Speech Recognition (ASR), Language Understanding Module, State Tracker and Dialog Policy together forming the Dialog Manager, the Natural Language Generator (NLG), also known as Response Generator. In this chapter we focus on data-driven dialog systems as well as interactive dialog systems in which human or a simulated human is involved in learning dialog system components using deep learning on real-world conversational dialogs.

Types of Tasks	Examples
Information consumption	*'what is the conference schedule'* *'which room is the talk in ?'*
Task Completion	*'set my alarm for 3pm tomorrow'* *'find me kid-friendly vegetarian restaurant in downtown Seattle'* *'schedule a meeting with sandy after lunch.'*
Decision Support	*'why are sales in south region far behind ?'*
Social Interaction (chit-chat)	*'how is your day going'* *'i am as smart as human ?'* *'i love you too.'*

Table 3.1: Type of tasks that dialog systems are currently used.

It should be noted that the spoken language or speech recognition component of in the overall spoken dialog systems has a huge impact on the success of the full systems. This front-end component involves several factors that make it difficult for machines to recognize speech. The analysis of continuous speech is a difficult task on its own as there is huge variability in the speech signal and there are no clear boundaries between words. For technical detail of such and many other difficulties in building spoken language systems, we refer readers to (Huang and Deng, 2010; Deng and Li, 2013; Li et al., 2014; Deng and Yu, 2015; Hinton et al., 2012; He and Deng, 2011).

The speech recognition component of spoken dialogue systems is often speaker independent and does not take into account that it is the same user during the whole dialogue. In the end-to-end spoken dialogue systems, the inevitable errors in speech recognition would make the language understanding component harder than when the input is text — free of speech recognition errors (He and Deng, 2013). In the long history of spoken language understanding research, the difficulties caused by speech recognition errors forced the domains of spoken language understanding to be substantially narrower than language understanding in text form (Tur and Deng, 2011). However, due to the recent huge success of deep learning in speech recognition in recent years (Yu and Deng, 2015; Deng, 2016), recognition errors have been dramatically reduced, leading to increasingly broader application domains in the current conversational understanding systems[6].

Most early goal-driven dialogue systems were primarily based on hand-crafted rules (Aust et al., 1995; Simpson and Eraser, 1993) which immediately followed machine learning techniques for all components of the dialog system (Tur and De Mori, 2011; Gorin et al., 1997). Most of these work formulate dialogue as a sequential decision making problem based on Markov decision processes. With the deep neural networks, especially the research in speech recognition, spoken language understanding (e.g,. Feed forward neural networks (Hastie et al., 2009), RNNs (Goller and Kchler, 1996) including LSTMs (Graves and Schmidhuber, 2005), and dialog modeling (e.g,. deep reinforcement learning methods) have showed incredible suc-

[6] We refer the reader to the "Deep Learning in Conversational Language Understanding" chapter in this book for more details in discussing this issue.

cess in robustness and coherency of the dialog systems (Wen et al., 2016b; Dhingra et al., 2016a; Lipton et al., 2016). On the other hand, most earlier non-goal oriented systems has used simple rules, topic models and modeled dialogue as a stochastic sequence of discrete symbols (words) using higher-order Markov chains. Only recently, deep neural network architectures trained on large-scale corpora have been investigated and promising results have been observed (Ritter et al., 2011; Vinyals and Le, 2015; Lowe et al., 2015a; Sordoni et al., 2015a; Serban et al., 2016b, 2017). One of the biggest challenges of non-goal oriented systems that use deep neural networks is that they require substantially large corpora in order to achieve good results.

This chapter is structured as follows. In the next in Section 3.2, a high-level overview of the deep learning tools that are used in building sub-components of the current dialogue systems are provided. Section 3.3 describes the individual system components of the goal oriented neural dialog systems and provide examples of recently presented research work. In Section 3.4, types of user simulators that are use deep learning technologies are discussed. Later methods on how deep learning methods are utilized in natural language generation are presented in Section 3.5. Later section delves into the deep learning methods that are relevant for building end-to-end dialogue systems in Section 3.6. In Section 3.7, the open-domain non-goal oriented dialog systems are presesented followed by the current datasets used to building deep dialog models and provide links to the each corpus in turn while emphasizing how the dialogues were generated and collected. Section 3.9 briefly touches on open source neural dialog system modeling software. Evaluating dialog systems and the measures used to evaluate them are presented in Section 3.10. Finally in Section 3.11, this chapter concludes with a survey of projections into the future of dialog modeling.

3.2 Learning Methology for Compoments of a Dialog System

In this section, we will summarize some of the deep learning techniques that are used in building conversational agents. Deep Learning technologies have been used to model nearly all of the components of the dialog systems. We investigate such methods below under three different categories: *discriminative*, *generative* and *decision making based*, specifically reinforcement learning.

3.2.1 Discriminative Methods

Deep learning methods that model the posterior $p(y|x)$ directly with abundance of supervised data has been one the most investigated approaches in dialog modeling research. Most advanced and prominent approaches has been investigated for the spoken language understanding (SLU) tasks such as goal estimation and intention

identification from users commands, which are essential components in spoken dialog systems and they are modeled as multi-output classification tasks. Most research work in this area use Deep Neural Networks for classification specifically multi-layered feed forward neural networks or multi-layered perceptrons (Hastie et al., 2009). These models are called feed-forward because information flows through the function being evaluated from x, through the intermediate computations used to define f, and finally to the output y.

Deep Structured Semantic Models (DSSM), or more general, Deep Semantic Similarity Models, are one of the approaches in deep learning research which is commonly used for multi/single class text classification thats which intrinsically learn similarities between two text while discovering latent features. In dialog system modeling, DSSM approaches that are mainly for SLU's classification tasks (Huang et al., 2013). DSSMs are a deep neural network (DNN) modeling technique for representing text strings (sentences, queries, predicates, entity mentions, etc.) in a continuous semantic space and modeling semantic similarity between two text strings (e.g., Sent2Vec). Also commonly used are the Convolutional Neural Networks (CNN) which utilize layers with convolving filters that are applied to local features (LeCun et al., 1998). Originally invented for computer vision, CNN models have subsequently been shown to be effective for SLU models mainly for learning latent features that are otherwise impossible to extract with standard (non-)linear machine learning approaches.

Semantic slot filling is one of the most challenging problems in SLU and is considered as as a sequence learning problem. Similarly, belief tracking or dialog state tacking are also considered sequential learning problems for the reasons that they mainly maintain the state of the dialog through each conversation in the dialog. Although CNNs are a great way to pool local information, they do not really capture the sequentiality of the data and not the first choice when it comes to sequential modeling. Hence to tackle sequential information in modeling user utterances in dialog systems, most research has focused on using Recurrent Neural Networks (RNN) which help tackle sequential information.

Memory Networks (Weston et al., 2015; Sukhbaatar et al., 2015; Bordes et al., 2017) are a recent class of models that have been applied to a range of natural language processing tasks, including question answering (Weston et al., 2015), language modeling (Sukhbaatar et al., 2015), etc. Memory networks in general work by first writing and then iteratively reading from a memory component (using hops) that can store historical dialogs and short-term context to reason about the required response. They have been shown to perform well on those tasks and to outperform some other end-to-end architectures based on Recurrent Neural Networks. Also attention based RNN networks such as Long-Short-Term-Memory Networks (LSTM) take different approach to keep the memory component and learn to attend dialog context (Liu and Lane, 2016a).

Obtain large corpora for every new application may not be feasible to build deep supervised learning models. For this reason the use of other related datasets can effectively bootstrap the learning process. Particularly in deep learning, the use of related datasets in pre-training a model is an effective method of scaling up to com-

plex environments (Kumar et al., 2015). This is crucial in open-domain dialogue systems, as well as multi task dialog systems (e.g., travel domain comprising of several tasks from different domains such as hotel, flight, restaurants, etc.). Dialog modeling researchers have already proposed various deep learning approaches for applying transfer learning to build data-driven dialogue systems such as learning sub-components of the dialogue system (e.g. intent and dialogue act classification) or learning end-to-end dialog system using transfer learning.

3.2.2 Generative Methods

Deep generative models have recently become popular due to their ability to model input data distributions and generate realistic examples from those distributions and in turn has recently entered in the dialog system modeling research field. Such approaches are largely considered in clustering objects and instances in the data, extracting latent features from unstructured text, or dimensionality reduction. A large portion of the category of dialog modeling systems that use deep generative models investigate open domain dialog systems specifically focusing on neural generative models for response generation. Common to these work are encoder-decoder based neural dialog models (see Figure 3.5) (Vinyals and Le, 2015; Lowe et al., 2015b; Serban et al., 2017; Shang et al., 2015), in which the encoder network used the entire history to encode the dialog semantics and the decoder generates natural language utterance (e.g., sequence of words representing systems' response to user's request). Also used are RNN based systems that map an abstract dialogue act into an appropriate surface text (Wen et al., 2015a).

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) is one topic in generative modeling which has very recently appeared in the dialog field as neural dialog modeling tasks specifically for dialog response generation. While Li et al. (2017) use deep generative adversarial networks for response generation, Kannan and Vinyals (2016) investigate the use of an adversarial evaluation method for dialogue models.

3.2.3 Decision Making

The key to a dialog system is its decision making module, which is also known as the dialogue manager or also referred to as dialogue policy. The dialog policy chooses system actions at each step of the conversation to guide the dialogue to successful task completion. The system actions include interacting with the user for getting specific requirements for accomplishing the task, as well as negotiating and offering alternatives. Optimization of statistical dialogue managers using *Reinforcement Learning* (RL) methods is an active and promising area of research (Fatemi et al., 2016a; Su et al., 2016; Fatemi et al., 2016b; Lipton et al., 2016; Shah et al., 2016;

Williams and Zweig, 2016a; Dhingra et al., 2016a). The RL setting fits the dialogue setting quite well because RL is meant for situations when feedback may be delayed. When a conversational agent carries a dialogue with a user, it will often only know at the end whether or not the dialogue was successful and the task was achieved.

Aside from the above categories, deep dialog systems has also been introduced with novel solutions involving applications of transfer learning and domain adaptation for next generation dialog systems, specifically focusing on domain transfer in spoken language understanding (Kim et al., 2017a,b, 2016b,a) and dialog modeling (Gai et al., 2016, 2015; Lipton et al., 2016).

3.3 Goal-Oriented Neural Dialogue Systems

The most useful applications of dialog systems can be considered to be the goal-oriented and transactional in which the system needs to understand a user request and complete a related task with a clear goal within a limited number of dialog turns. We will provide description and recent related work for each component of goal oriented dialog systems in detail.

3.3.1 Neural Language Understanding

With the power of deep learning, there is increasing research work focusing on applying deep learning for language understanding. In the context of goal-oriented dialogue systems, language understanding is tasked with interpreting user utterances according to a semantic meaning representation, in order to enable with the back-end action or knowledge providers. Three key tasks in such targeted understanding applications are domain classification, intent determination and slot filling (Tur and De Mori, 2011), aiming to form a semantic frame that captures the semantics of user utterances/queries. Domain classification is often completed first in spoken language understanding (SLU) systems, serving as a top-level triage for subsequent processing. Intent determination and slot filling are then run for each domain to fill a domain specific semantic template. An example semantic frame for a movie-related utterance, *find recent comedies by James Cameron*, is shown in Figure 3.2.

With the advances on deep learning, deep belief networks (DBNs) with deep neural networks (DNNs) have been applied to domain and intent classification tasks (Sarikaya et al., 2011; Tur et al., 2012; Sarikaya et al., 2014). More recently, Ravuri and Stolcke (2015) proposed an RNN architecture for intent determination, where an encoder network is first estimates a representation for the input utterance, and then a single step decoder estimates a domain/intent class for the input utterance using a single step decoder network.

For slot filling, deep learning has been viewed as a feature generator and the neural architecture can be merged with CRFs (Xu and Sarikaya, 2013). Yao et al. (2013)

W	find	recent	comedies	by	james	cameron
	↓	↓	↓	↓	↓	↓
S	O	B-date	B-genre	O	B-dir	I-dir
D	movies					
I	find_movie					

Fig. 3.2: An example utterance with annotations of semantic slots in IOB format (S), domain (D), and intent (I), B-dir and I-dir denote the director name. (Image from the published paper (Hakkani-Tür et al., 2016))

and Mesnil et al. (2015) later employed RNNs for sequence labeling in order to perform slot filling. Recent studies focused on sequence to sequence models (Kurata et al., 2016), sequence to sequence models with attention (Simonnet et al., 2015), multi-domain training (Jaech et al., 2016), multi-task training (Tafforeau et al., 2016), multi-domain joint semantic frame parsing (Hakkani-Tür et al., 2016; Liu and Lane, 2016b), and context modeling using end-to-end memory networks (Chen et al., 2016; Bapna et al., 2017). These will be described in more detail in the language understanding chapter.

3.3.2 Dialog State Tracker

The next step in spoken dialog systems pipeline is dialog state tracking (DST), which aims to track system's belief on user's goal through the course of a conversation. The dialog state is used for querying the back-end knowledge or information sources and for determining the next state action by the dialog manager. At each turn in a dialog, DST gets as input the estimated dialogue state from the previous user turn, s_{t-1}, and the most recent system and user utterances and estimates the dialog state s_t for the current turn. In the past few years, the research on dialog state tracking has accelerated owing to the data sets and evaluations performed by the dialog state tracking challenges (Williams et al., 2013; Henderson et al., 2014). The state-of-the-art dialog managers focus on monitoring the dialog progress by neural dialog state tracking models. Among the initial models are the RNN based dialog state tracking approaches (Henderson et al., 2013) that has shown to outperform Bayesian networks (Thomson and Young, 2010). More recent work on Neural Dialog Managers that provide conjoint representations between the utterances, slot-value pairs as well as knowledge graph representations (Wen et al., 2016b; Mrkšić et al., 2016) demonstrate that using neural dialog models can overcome current obstacles of deploying dialogue systems in larger dialog domains.

3.3.3 Deep Dialog Manager

A dialog manager is a component of a conversational dialog system, which interacts in a natural way to help the user complete the tasks that the system is designed to support. It is responsible for the state and flow of the conversation, hence determines what policy should be used. The input to the dialog manager is the human utterance, which is converted to some system-specific semantic representation by the natural language understanding component. For example, in a flight-planning dialog system, the input may look like "ORDER(from=SFO,to=SEA,date=2017-02-01)". The dialog manager usually maintains state variables, such as the dialog history, the latest unanswered question, the recent user intent and entities, etc., depending on the domain of the dialog. The output of the dialog manager is a list of instructions to other parts of the dialog system, usually in a semantic representation, for example "Inform(flight-num=555,flight-time=18:20)". This semantic representation is converted into natural language by the natural language generation component.

Typically, an expert manually designs a dialog management policy and incorporates several dialog design choices. Manual dialog policy design is intractable and does not scale as the performance of the dialog policy depends on several factors including domain specific features, robustness of the automatic speech recognizer (ASR) system, the task difficulty, to name a few. Instead of letting a human expert write a complex set of decision rules, it is more common to use reinforcement learning. The dialog is represented as a Markov Decision Process (MDP) - a process where, in each state, the dialog manager has to select an action, based on the state and the possible rewards from each action. In this setting, the dialog author should only define the reward function, for example: in restaurant reservation dialogs, the reward is the user success in reserving a table successfully; in information seeking dialogs, the reward is positive if the human receives the information, but there is also a negative reward for each dialog step. Reinforcement learning techniques are then used to learn a policy, for example, what type of confirmation should the system use in each state (Lemon and Rieserr, 2009). A different way to learn dialog policies is to try to imitate humans, using Wizard of Oz experiments, in which a human sits in a hidden room and tells the computer what to say (Passonneau et al., 2011).

For complex dialogue systems, it is often impossible to specify a good policy a priori and the dynamics of an environment may change over time. Thus, learning policies on-line and interactively via reinforcement learning has emerged as a popular approach (Singh et al., 2016; Gasic et al., 2010; Fatemi et al., 2016b). For instance, the ability to compute an accurate reward function is essential for optimizing a dialogue policy via reinforcement learning. In real-world applications, using explicit user feedback as the reward signal is often unreliable and costly to collect. Su et al. (2016) propose an on-line learning framework in which the dialogue policy is jointly trained alongside the reward model via active learning with a Gaussian process model. They propose a three main system components including dialogue policy, dialogue embedding creation, and reward modeling based on user feedback (see Figure 3.3). They use episodic turn-level features extracted from a dialogue

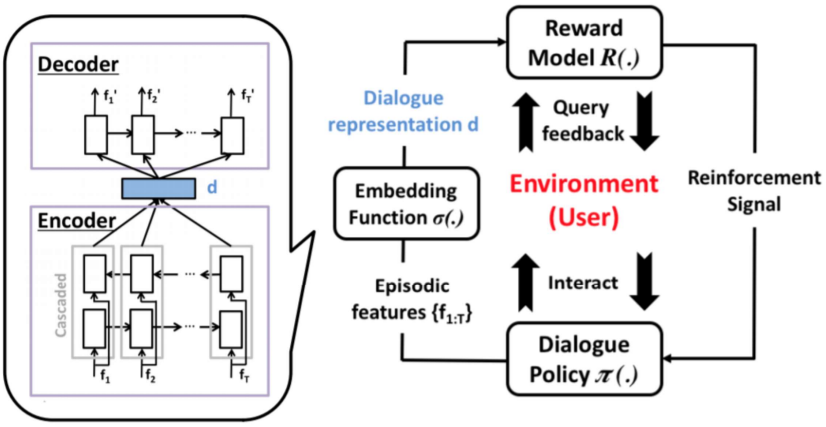

Fig. 3.3: Schematic of the dialog policy learning with deep encoder-decoder networks. The three main system components dialogue policy, dialogue embedding creation, and reward modeling based on user feedback. (Image from the published paper (Su et al., 2016))

and build a Bi-directional Long Short-Term Memory network (BLSTM) for their dialogue embedding creation.

Efficient dialog policy learning with deep learning technologies has recently been the focus of dialog researcher with the recent advancements in deep reinforcement learning. For instance Lipton et al. (2016) investigate understanding boundaries of the deep neural network structure of the dialog policy model to efficiently explore different trajectories via Thompson sampling, drawing Monte Carlo samples from a Bayesian neural network (Blundell et al., 2015). They use deep Q-network to optimize the policy. They explore a version of their approach that incorporates the intrinsic reward from Variational Information Maximizing Exploration (VIME) (Blundell et al., 2015). Their Bayesian approach addresses uncertainty in the Q-value given the current policy, whereas VIME addresses uncertainty of the dynamics of under-explored parts of the environment. Thus there is a synergistic effect of combining the approaches. On the domain extension task, the combined exploration method proved especially promising, outperforming all other methods.

There are several other aspects that affect the policy optimization for dialog managers. Some of which include learning policies under multi-domain systems (Gasic et al., 2015; Ge and Xu, 2016), committee based learning for multi-domain systems (Gasic et al., 2015), learning domain independent policies (Wang et al., 2015), adapting to grounded word meanings (Yu et al., 2016), adapting to new user behaviors (Shah et al., 2016), to name a few. Among these systems Peng et al. (2017) investigate hierarchal policy learning for task oriented systems that has composite sub-tasks. This domain particularly challenging and the authors tackle with the

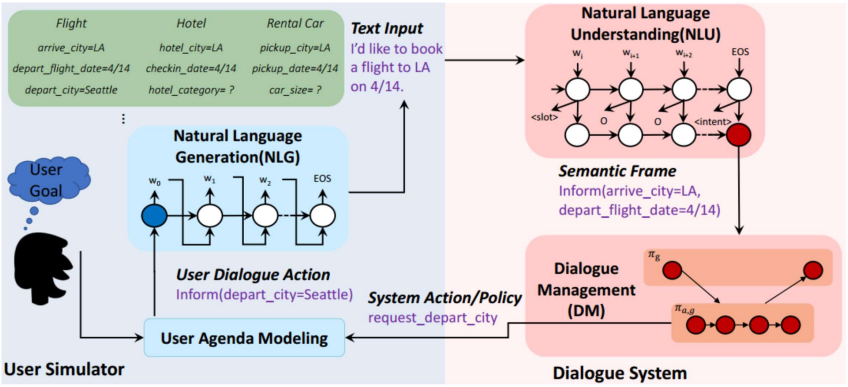

Fig. 3.4: Illustration of the Composite Task Completion Dialog System (Image from the published paper (Peng et al., 2017))

issue of reward sparsity, satisfying slot constraints across subtasks. This requirement makes most of the existing methods of learning multi-domain dialogue agents (Cuayahuitl et al., 2016; Gasic et al., 2015) inapplicable: these methods train a collection of policies, one for each domain, and there is no cross-domain constraints required to successfully complete a dialogue. As shown in Figure 3.4 their composite task-completion dialogue agent consists of four components: (1) an LSTM-based language understanding module for identifying user intents and extracting associated slots; (2) a state tracker for tracking the dialogue state; (3) a dialogue policy which selects the next action based on the current state; and (4) a model-based natural language generator for converting agent actions to natural language responses. Following the options over MDP's formalism (Sutton and Singh, 1999) they build their agent to learn a composite task such as travel planning, subtasks like book-flight-ticket and reserve-hotel which can be modeled as options.

3.4 Model-Based User Simulators

User simulators for spoken dialogue systems aim at generating artificial interactions supposed to be representative of what would be an actual dialogue between a human user and a given dialogue system. Model-based simulated users for building dialog models is not as common as the other components of the dialog systems; detailed reviews of some of these methods are presented in (Schatzmann et al., 2006; K. et al., 2005, 2006). In this section we only investigate deep learning methods for user simulation, that is, methods purely based on data and deep learning approaches models.

The early spoken dialogue systems optimization required a lot of data because of inefficiency of reinforcement learning algorithms, justifying the use of simulation. In recent years, sample efficient reinforcement learning methods were applied to spoken dialogue systems optimization. With this, models can be trained to learn optimal dialogue strategies directly from large amounts of data collected even from suboptimal systems with actual users (Li et al., 2009; Pietquin et al., 2011b) but also from online interactions (Pietquin et al., 2011a). This makes it much more appealing for the dialog systems to be trained using a simulated user with user feedback and corrected as the process continues.

There are several reasons that make learning parameters of a user simulation model hard to optimize because most of the system features are hidden (e.g., user goal, mental states, dialog history, etc.). Focusing on this problem, Asri et al. (2016) presented a sequence-to-sequence base user simulator on non-goal oriented domains (e.g,. chit-chat) that takes into account the entire dialogue history. Their user simulator does not rely on any external data structure to ensure coherent user behavior, and it does not require mapping to a summarized action space, which makes it able to model user behavior with finer granularity.

Crook and Marin (2017) explore sequence to sequence learning approach for NL-to-NL simulated user models for goal oriented dialog systems. They present several extensions to their architecture to incorporate context in different ways and investigate the efficacy of each method in comparison to language modeling baseline simulator on a personal assistant system domain. Their findings showed that context based sequence-to-sequence method can generate human like utterances outperforming all other baselines.

3.5 Natural Language Generation

Natural language generation (NLG) is the process of generating text from a meaning representation. It can be taken as the reverse of the natural language understanding. NLG systems provide a critical role for text summarization, machine translation and dialog systems. While several general-purpose rule-based generation systems have been developed (Elhadad and Robin, 1996), they are often quite difficult to adapt to small, task-oriented applications because of their generality. To overcome this, several people have proposed different solutions. Bateman and Henschel (1999) have described a lower cost and more efficient generation system for a specific application using an automatically customized sub-grammar. Busemann and Horacek (1998) describe a system that mixes templates and rule-based generation. This approach takes advantages of templates and rule-based generation as needed by specific sentences or utterances. Stent (1999) has also proposed a similar approach for a spoken dialog system. Although such approaches are conceptually simple and tailored to the domain, so often good quality, they lack generality (e.g, repeatedly encode linguistic rules such as subject-verb agreement), have little variation in style and difficult to grow and maintain (e.g., usually each new utterance is added by hand).

Such approaches impose the requirement of writing grammar rules and acquiring the appropriate lexicon, which requires a specialist activity.

Machine learning based (trainable) NLG systems are more common in today's dialog systems. Such NLG systems use several sources as input such as: *content plan*, representing meaning representation of what to communicate with the user (e.g., describe a particular restaurant), *knowledge-base*, structured database to return domain specific entities, (e.g., database of restaurants), *user model*, a model that imposes constraints on output utterance (e.g, user wants short utterances), *dialog history*, the information from previous turns to avoid repetitions, referring expressions, etc. The goal is to use these meaning representations that indicate what to say (e.g, entities described by features in an ontology) to output natural language string describing the input (e.g., *zucca's food is delicious.*).

Trainable NLG systems can produce various candidate utterances (e.g., stochastically or rule base) and use a statistical model to rank them (Dale and Reiter, 2000). The statistical model assigns scores to each utterance and is learnt based on textual data. Most of these systems use bi-gram and tri-gram language models to generate utterances. The trainable generator approach exemplified by the HALOGEN (Langkilde and Knight, 1998) and SPaRKy system (Stent et al., 2004) are among the most notable trainable approaches. These systems include various trainable modules within their framework to allow the model to adapt to different domains (Walker et al., 2007), or reproduce certain style (Mairesse and Walker, 2011). However, these approaches still require a handcrafted generator to define the decision space. The resulting utterances are therefore constrained by the predefined syntax and any domain-specific colloquial responses must be added manually. In addition to these approaches, corpus-based methods (Oh and Rudnicky, 2000; Mairesse and Young, 2014; Wen et al., 2015a) have been shown to have flexible learning structures with the goal of learning generation directly from data by adopting an over-generation and re-ranking paradigm (Oh and Rudnicky, 2000), in which final responses are obtained by re-ranking a set of candidates generated from a stochastic generator.

With the advancement of deep neural network systems, more sophisticated NLG systems can be developed that can be trained from un-aligned data or produce longer utterances. Recent study have shown that especially with the RNN methods (e.g., LSTMs, GRUs, etc.) more coherent, realistic and proposer answers can be generated. Among these studies, Vinyals and Le (2015)'s work on Neural Conversational Model has opened a new chapter in using encoder-decoder based models for generation. Their model is based on two LSTM layers. One for encoding the input sentence into a "thought vector", and another for decoding that vector into a response. This model is called Sequence-to-sequence or seq2seq. The model only gives simple and short answers to questions.

Sordoni et al. (2015b) propose three neural models to generate a response (r) based on a context and message pair (c,m). The context is defined as a single message. They propose several models, the first one of which is a basic Recurrent Language Model that is fed the whole (c,m,r) triple. The second model encodes context and message into a BoW representation, put it through a feed-forward neural network encoder, and then generates the response using an RNN decoder. The last

model is similar but keeps the representations of context and message separate instead of encoding them into a single BoW vector. The authors train their models on 29M triple data set from Twitter and evaluate using BLEU, METEOR and human evaluator scores. Because (c,m) is very long on average the authors expect their first model to perform poorly. Their model generates responses degrade with length after 8 tokens.

Li et al. (2016b) present a method which adds coherency to the response generated by sequence-to-sequence models such as the Neural Conversational Model (Vinyals and Le, 2015). They define persona as the character that an agent performs during conversational interactions. Their model combines identity, language, behavior and interaction style. Their model may be adapted during the conversation itself. Their proposed models yields performance improvements in both perplexity and BLEU scores over baseline sequence-to-sequence models. Compared to Persona based Neural Conversational Model, the baseline Neural Conversational Model fails to maintain a consistent persona throughout the conversation resulting in incoherent responses. A similar approach in (Li et al., 2016a) uses a Maximum Mutual Information (MMI) objective function to generate conversational responses. They still train their models with maximum likelihood, but use MMI to generate responses during decoding. The idea behind MMI is that it promotes more diversity and penalizes trivial responses. The authors evaluate their method using BLEU scores, human evaluators, and qualitative analysis and find that the proposed metric indeed leads to more diverse responses.

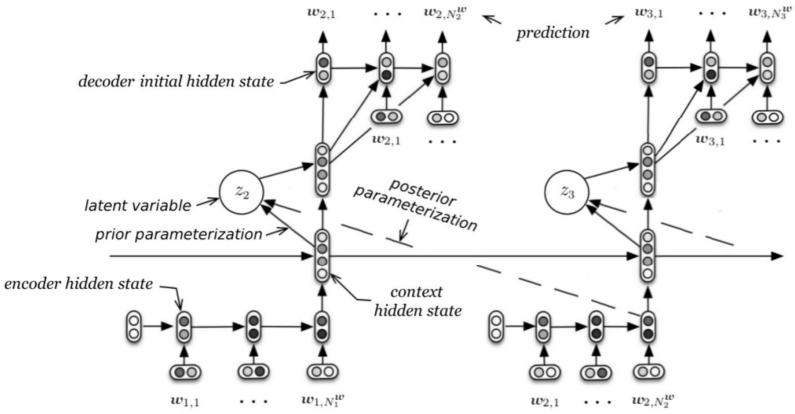

Fig. 3.5: Hierarchal Encoder-Decoder Model computational graph. Diamond boxes represent deterministic variables and rounded boxes represent stochastic variables. Full lines represent the generative model and dashed lines represent the approximate posterior model. (Image from the published paper (Serban et al., 2017))

Serban et al. (2017) present a hierarchical latent variable encoder-decoder model for generating dialogues. Their goal is to generate natural language dialog responses. Their model assumes that each output sequence can be modeled in a two-level hierarchy: sequences of sub-sequences, and sub-sequences of tokens. For example, a dialogue may be modeled as a sequence of utterances (sub-sequences), with each utterance modeled as a sequence of words. Given this, their model consists of three RNN modules: an encoder RNN, a context RNN and a decoder RNN. Each sub-sequence of tokens is deterministically encoded into a real-valued vector by the encoder RNN. This is given as input to the context RNN, which updates its internal hidden state to reflect all information up to that point in time. The context RNN deterministically outputs a real-valued vector, which the decoder RNN conditions on to generate the next sub-sequence of tokens (see Figure 3.5).

Recent work in natural language generation has focused on using reinforcement learning strategies to explore different learning signals (He et al., 2016; Williams and Zweig, 2016b; Wen et al., 2016a; Cuayhuitl, 2016). The motivation for this renewed interest in reinforcement learning stems from issues of using teacher forcing for learning. Text generation systems trained using word-by-word cross-entropy loss with gold sequences as supervision have produced locally coherent generations, but generally fail to capture the contextual dynamics of the domain they are modeling. Recipe generation systems that are conditioned on their ingredients and recipe title, for example, do not manage to combine the starting ingredients into their end dish in a successful way. Similarly, dialogue generation systems often fail to condition their responses on previous utterances in the conversation. Reinforcement learning allows models to be trained with rewards that go beyond predicting the correct word. Mixing reward schemes using teacher forcing and other more global metrics has recently become a popular for producing more domain-relevant generations.

3.6 End-to-end Deep Learning Approaches to Building Dialogue Systems

End-to-end dialog systems are considered a cognitive system, which has to carry out natural language understanding, reasoning, decision making and natural language generation within the same network in order to replicate or emulate the behavior of the agents in the training corpus. This has not been fully investigated before the deep learning technologies being used for dialog system building. Now, building such systems with today's deep learning technologies are much easier because of the fact that with the deep learning systems and back-propagation all parameters can be trained jointly. In the next we will briefly investigate the recent end-to-end dialog models for goal and non-goal oriented systems.

One of major obstacles in building and end-to-end goal oriented dialog systems is that the the database calls made by the system to retrieve the information requested by the user is not differentiable. Specifically, the query generated by the system and sent to knowledge base is done in a manual way, which means that part of the system

is not trained and no function is learnt. This cripples the deep learning model into incorporating the knowledge base response and the information it receives. Also, the neural response generation part is trained and run as separate of the dialog policy network. Putting all this together, training the whole cycle end-to-end has not been fully investigated until recently.

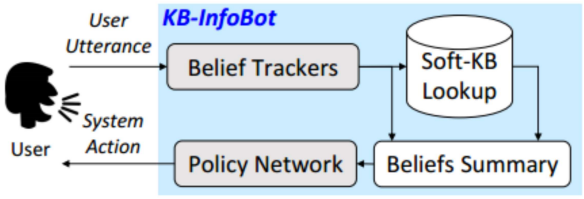

Fig. 3.6: High-level overview of the end-to-end Knowledge-Base-InfoBot: a multi-turn dialogue agent which helps users search Knowledge Bases without composing complicated queries. Such goal-oriented dialogue agents typically need to interact with an external database to access real-world knowledge. This model replaces symbolic queries with an induced soft posterior distribution over the knowledge base that indicates which entities the user is interested in. The components with trainable parameters are highlighted in gray. (Image from the published paper (Dhingra et al., 2016b))

Recently, there has been a growing body of literature focusing on building end-to-end dialogue systems, which combine feature extraction and policy optimization using deep neural networks. Wen et al. (2015b) introduced a modular neural dialogue agent, which uses a hard knowledge base lookup, thus breaking the differentiability of the whole system. As a result, training of various components of the dialogue system is performed separately. The intent network and belief trackers are trained using supervised labels specifically collected for them; while the policy network and generation network are trained separately on the system utterances.

Dhingra et al. (2016b) introduce a modular approach, consisting of: a belief tracker module for identifying user intents, extracting associated slots, and tracking the dialogue state; an interface with the database to query for relevant results (Soft-KB lookup); a summary module to summarize the state into a vector; a dialogue policy which selects the next system action based on current state and a easily configurable template-based Natural Language Generator (NLG) for converting dialogue acts into natural language (see Figure 3.6). The main contribution of their work is that it retains modularity of the end-to-end network by keeping the belief trackers separate, but replaces the hard lookup with a differentiable one. They propose a differentiable probabilistic framework for querying a database given the agents beliefs over its fields (or slots). showing that the downstream reinforcement learner can discover better dialogue policies by providing it more information.

The non-goal oriented end-to-end dialog systems investigate the task of building open domain, conversational dialogue systems based on large dialogue corpora. Serban et al. (2015) incorporate generative models to produce system responses that are autonomously generated word-by-word, opening up the possibility for realistic, flexible interactions. They demonstrate that a hierarchical recurrent neural network generative model can outperform both n-gram based models and baseline neural network models on the task of modeling utterances and speech acts.

3.7 Deep Learning for Open Dialog Systems

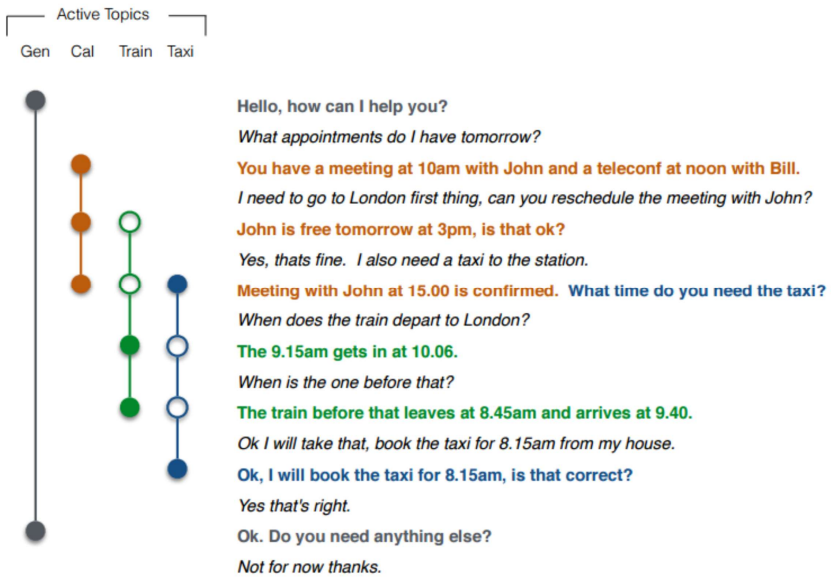

Fig. 3.7: Open Dialog Dialog system conversational dialog example. (Image from the Steve Young's SLSP2015 slides (Young, 2015))

Open domain dialog systems, also known as non-task-oriented systems, do not have a stated goal to work towards. These type of dialog systems are mainly useful for interactions in social environments (e.g, social bots) as well as many other useful scenarios (e.g., keeping elderly people company (Higashinaka et al., 2014), or entertaining users (Yu et al., 2015), to name a few. Open-domain spoken dialogue systems support a natural conversation about any topic within a wide cover-

age Knowledge Graph (KG). The KG can contain not only ontological information about entities but also the operations that might be applied to those entities (e.g. find flight information, book a hotel room, buy an ebook, etc.) (see example dialog in Figure 3.7).

The non-task oriented systems do not have a goal, nor have a set of states or slots to follow however they do have intentions. Due to this, there has been several work on non-goal oriented dialog systems that focus preliminarily on response generation which use dialog history (human-agent conversations) as input to propose a response to the user. Among these work are machine translation (Ritter et al., 2011), retrieval-based response selection (Banchs and Li., 2012), and sequence-to-sequence models with different structures, such as, vanilla recurrent neural networks (Vinyals and Le, 2015), hierarchical neural models (Serban et al., 2015, 2016a; Sordoni et al., 2015b; Shang et al., 2015), and memory neural networks (Dodge et al., 2015). There are several motivations for developing non-goal-driven systems. They may be deployed directly for tasks which do not naturally exhibit a directly measurable goal (e.g. language learning) or simply for entertainment. Also if they are trained on corpora related to the task of a goal-driven dialogue system (e.g. corpora which cover conversations on similar topics) then these models can be used to train a user simulator, which can then train the policy strategy.

Until very recently there has been no research on combining the goal oriented and non-goal oriented dialog systems. In a recent work, a first attempt to create a framework that combines these two types of conversations in a natural and smooth manner for the purpose of improving conversation task success and user engagement is presented (Yu et al., 2017). Such a framework is especially useful to handle users who do not have explicit intentions.

3.8 Datasets for Dialog Modeling

In the last years, there has been several publicly available conversational dialog dataset released. Dialog corpora may vary based on several characteristics of the conversational dialog systems. Dialog corpora can be classified based on written, spoken or multi-model properties, or human-to-human or human-to-machine conversations, or natural or un-natural conversations (e.g., in a Wizard-of-Oz system, a human thinks (s)he is speaking to a machine, but a human operator is in fact controlling the dialogue system). In this section we provide a brief overview of these publicly available datasets that are used by the community, for spoken language understanding, state tracking, dialog policy learning, etc. specifically for task completion task. We leave out for open-ended non-task completion datasets in this section.

3.8.1 The Carnegie Mellon Communicator Corpus

This corpus contains human-machine interactions with a travel booking system. It is a medium-sized dataset of interactions with a system providing up-to-the-minute flight information, hotel information, and car rentals. Conversations with the system were transcribed, along with the users comments at the end of the interaction

3.8.2 ATIS - Air Travel Information System Pilot Corpus

The ATIS (Air Travel Information System) Pilot Corpus (Hemphill et al., 1990) is one of the first human-machine corpora. It consists of interactions, lasting about 40 minutes each, between human participants and a travel-type booking system, secretly operated by humans. Unlike the Carnegie Mellon Communicator Corpus, it only contains 1041 utterances.

3.8.3 Dialog State Tracking Challenge Dataset

The Dialog State Tracking Challenge (DSTC) is an on-going series of research community challenge tasks. Each task released dialog data labeled with dialog state information, such as the users desired restaurant search query given all of the dialog history up to the current turn. The challenge is to create a tracker that can predict the dialog state for new dialogs. In each challenge, trackers are evaluated using held-out dialog data. Williams et al. (2016) provide an overview of the challenge and datasets which we summarize below:

DSTC1[7]. This dataset used human-computer dialogs in the bus timetable domain. Results were presented in a special session at SIGDIAL 2013.

DSTC2 and DSTC3[8]. DSTC2 used human-computer dialogs in the restaurant information domain. DSTC2 uses large number of training dialog related to restaurant search. It has changing user goals, tracking 'requested slots'. Results were presented in special sessions at SIGDIAL 2014 and IEEE SLT 2014. DSTC3 is in tourist information domain which addressed the problem of adaptation to a new domain. DSTC2 and 3 were organized by Matthew Henderson, Blaise Thomson, and Jason D. Williams.

DSTC4[9]. The focus of this challenge is on a dialog state tracking task on human-human dialogs. In addition to this main task, a series of pilot tracks is introduced for the core components in developing end-to-end dialog systems based on the same dataset. Results were presented at IWSDS 2015. DSTC4 was organized by

[7] https://www.microsoft.com/en-us/research/event/dialog-state-tracking-challenge/

[8] http://camdial.org/ mh521/dstc/

[9] http://www.colips.org/workshop/dstc4/

Seokhwan Kim, Luis F. DHaro, Rafael E Banchs, Matthew Henderson, and Jason D. Williams.

DSTC5[10]. DSTC5 used human-human dialogs in the tourist information domain, where training dialogs were provided in one language, and test dialogs were in a different language. Results are presented in a special session at IEEE SLT 2016. DSTC5 was organized by Seokhwan Kim, Luis F. DHaro, Rafael E Banchs, Matthew Henderson, Jason D. Williams, and Koichiro Yoshino.

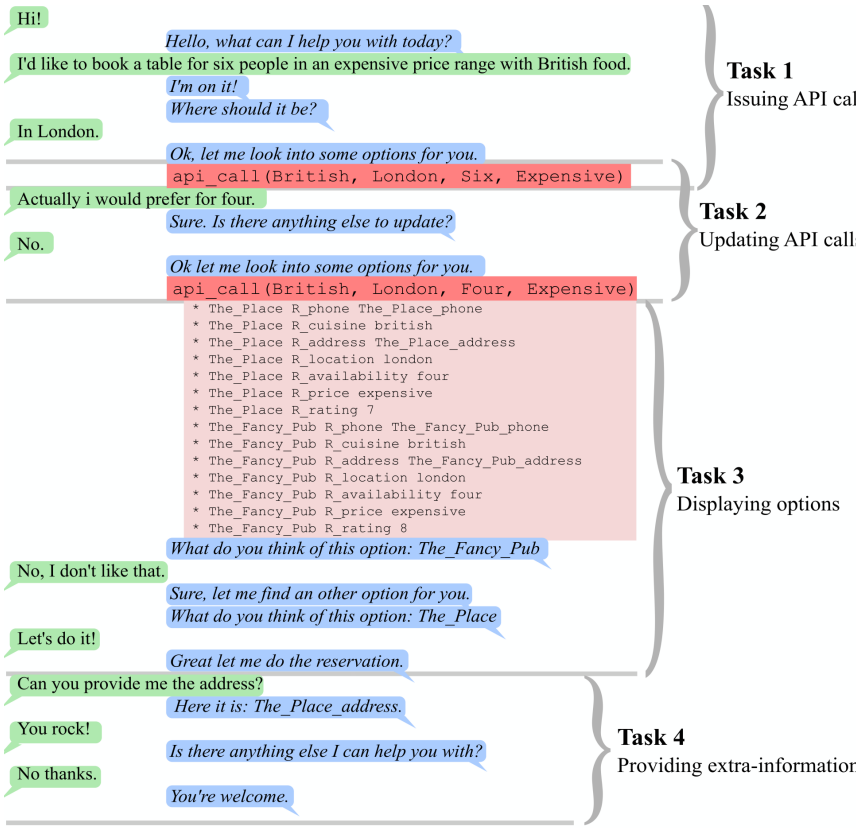

Fig. 3.8: A sample dialogue between a virtual agent and a customer, in restaurant domain.

[10] http://workshop.colips.org/dstc5/

3.8.4 Maluuba Frames Dataset

Frames [11] is presented to for research in conversational agents which can support decision-making in complex settings, i.e., booking a vacation including flights and a hotel. With this dataset the goal is to teach conversational agents that can help users explore a database, compare items, and reach a decision. The human-human conversation Frames data is collected using Wizard-of-Oz which is designed for composite task-completion dialogue setting. we consider an important type of complex task, called *composite task*, which consists of a set of subtasks that need to be fulfilled collectively. For example, in order to make a travel plan, the user first needs to book air tickets, reserve a hotel, rent a car, etc. in a collective way so as to satisfy a set of cross-subtask constraints, which are called *slot constraints*. Examples of slot constraints for travel planning are: hotel check-in time should be later than the departure flight time, hotel check-out time may be earlier than the return flight depart time, the number of flight tickets equals to that of hotel check-in people, and so on.

3.8.5 Facebook's Dialog Datasets

In the last year, Facebook AI and Research (FAIR) has released task oriented dialog datasets to be used by the dialog research community (Bordes et al., 2017) [12]. The objective of their project is to explore neural network architectures for question answering and goal oriented dialog systems. They designed a set of five tasks within the goal-oriented context of restaurant reservation (see example in Figure 3.8) Grounded with an underlying KB of restaurants and their properties (location, type of cuisine, etc.), these tasks cover several dialog stages and test if models can learn various abilities such as performing dialog management, querying KBs, interpreting the output of such queries to continue the conversation or dealing with new entities not appearing in dialogs from the training set.

3.8.6 Ubuntu Dialog Corpus

The Ubuntu Dialogue Corpus Lowe et al. (2015b) [13] consists of almost one million two-person conversations extracted from the Ubuntu chat logs about technical support for various Ubuntu-related problems. The dataset targets a specific technical support domain. Therefore it can be used as a case study for the development of AI agents in targeted applications, in contrast to chatbox systems. All conversations are

[11] https://datasets.maluuba.com/Frames

[12] https://github.com/facebookresearch/ParlAI

[13] https://github.com/rkadlec/ubuntu-ranking-dataset-creator

carried out in text form (not audio). The dataset is orders of magnitude larger than structured corpora such as those of the DSTC. Each conversation in their dataset includes several turns, as well as long utterances.

3.9 Open Source Dialog Software

Conversational dialog systems has been the focus of many leading companies and researchers in the field have been building systems to improve several components of the conversational dialog systems. Some work just focus on proving trainable datasets and labeling platforms, or machine learning algorithms that can learn through interaction, others provide environment (simulators) to train interactive dialog systems. Below we briefly summarize the open source software/platforms that are readily accessible for dialog researchers.

- **OpenDial** [14]: The toolkit has been originally developed by the Language Technology Group of the University of Oslo (Norway), with Pierre Lison as main developer. It is a Java-based, domain-independent toolkit for developing spoken dialogue systems. OpenDial provides a tool to build full-fledged, end-to-end dialogue system, integrating speech recognition, language understanding, generation and speech synthesis. The purpose of OpenDial is to combine the benefits of logical and statistical approaches to dialogue modeling into a single framework. The toolkit relies on probabilistic rules to represent the domain models in a compact and human-readable format. Supervised or reinforcement learning techniques can be applied to automatically estimate unknown rule parameters from relatively small amounts of data (Lison, 2013). The tool also enables to incorporate expert knowledge and domain-specific constraints in a robust, probabilistic framework.
- **ParlAI** Along with the datasets, Facebook AI and Research (FAIR) have released a platform entitled ParlAI[15] with the goal of providing researchers a unified framework for training and testing dialog models, multi-task training over many datasets at once as well as seamless integration of Amazon Mechanical Turk for data collection and human evaluation.
- **Alex Dialog Systems Framework** [16]: This is a dialog systems framework that facilitates research into and development of spoken dialogue system. It is provided by a group at UFAL[17] - the Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics, Charles University in Prague, Czech Republic. The tool provides baseline components that is required for a building spoken dialogue systems as well as provides additional tools for processing dia-

[14] https://github.com/plison/opendial

[15] https://github.com/facebookresearch/ParlAI

[16] https://github.com/UFAL-DSG/alex

[17] http://ufal.mff.cuni.cz/

logue system interactions logs, e.g. for audio transcription, semantic annotation, or spoken dialog system evaluation.

- **SimpleDS**: This is a simple deep reinforcement learning dialog system[18] that enables training dialogue agents with as little human intervention as possible. It includes the Deep Q-Learning with experience replay (Mnih et al., 2013) and provides support for multi-threaded and client-server processing, and fast learning via constrained search spaces.
- **Cornell Movie Dialogs Corpus**: This corpus contains a large metadata-rich collection of fictional conversations extracted from raw movie scripts (Mizil and Lee, 2011). It contains several conversational exchanges between pairs of movie characters.
- **Others**: There are numerous software applications (some open sourced) that also provide non-task oriented dialog systems, e.g., chit-chat dialog systems. Such systems provide machine learning tools and conversational dialog engine for creating chat bots. Examples include **Chatterbot** [19], a conversational dialog engine for creating chat bots, **chatbot-rnn** [20], a toy chat-bot powered by deep learning and trained on data from Reddit, to name a few. In metaguide.com [21] top 100 chatbots are listed.

3.10 Dialog System Evaluation

Throughout this chapter we have been investigated several types of dialog models, i.e., task oriented which are considered domain dependent as well as open domain dialog software, which are semi-domain dependent which can open ended or can switch back and froth between task oriented and open domain conversational dialogs.

The task oriented dialog systems, which are typically component base, are evaluated based on the performance of each individual component. For instance, the CLU is evaluated based on the performance of the intent detection model, the slot sequence tagging models (Hakkani-Tür et al., 2016; Celikyilmaz et al., 2016; Tur and De Mori, 2011; Chen et al., 2016), etc., whereas the dialog state tracker is evaluated based on the accuracy of the state changes discovered during the dialog turns. The dialog policy for task oriented systems are typically evaluated based on the success rate of the completed task judged by either user or the real human. Typically, evaluation is done using human-generated supervised signals, such as a task completion test or a user satisfaction score. Also the length of the dialog has played role in shaping the dialog policy (Schatzmann et al., 2006).

[18] https://github.com/cuayahuitl/SimpleDS

[19] https://github.com/gunthercox/ChatterBot

[20] https://github.com/pender/chatbot-rnn

[21] http://meta-guide.com/software-meta-guide/100-best-github-chatbot

The real problem in evaluating the dialog models performance arises when the dialog systems are open domain. Most approaches focus on evaluating the dialogue response generation systems, which are trained to produce a reasonable utterance given a conversational context. This is a very challenging task since automatically evaluating language generation models is intractable to the availability of possibly very large set of correct answers. Nevertheless, today, several performance measures are used to automatically evaluate how appropriate the proposed response is to the conversation (Liu et al., 2016). Most of these metrics compare the generated response to the ground truth response of the conversation using word based similarity metrics and word-embedding based similarity metrics. Below we will summarize some of the metrics that are most commonly used in the dialog systems:

BLEU (Papineni et al., 2002) is an algorithm for evaluating the quality of text by investigating the co-occurrences of n-grams in the ground truth sequence (text) and the generated responses. BLEU uses a modified form of precision to compare a candidate translation against multiple reference translations:

$$P_n(r,\hat{r}) = \frac{\sum_k \min(h(k,r), h(k,\hat{r}_i))}{\sum_k h(k,r_i)}$$

where k represents all possible n-grams and $h(k,r)$ is the number of n-grams k in r. The metric modifies simple precision since text generation systems have been known to generate more words than are in a reference text. Such a score would favor shorter sequences. To remedy that, in (Papineni et al., 2002) a brevity score is used which yields BLUE-N score, where N is the maximum length of the n-grams and is defined as :

$$BLEU\text{-}N = b(r,\hat{r}) \exp(\sum_{n=1}^{N}) \beta_n \log P_n(r,\hat{r}))$$

where β_n is the weight factor and $b(\cdot)$ is the brevity penalty.

METEOR (Banerjee and Lavie, 2005) is another method which is based on BLEU and is introduced to address several weaknesses of BLEU. As with BLEU, the basic unit of evaluation is the sentence, the algorithm first creates an alignment between the reference and candidate generated sentences. The alignment is a set of mappings between unigrams and has to comply with several constraints including the fact that every unigram in the candidate translation must map to zero or one unigram in the reference followed by WordNet synonym matching, stemmed tokens and paraphrases of text. The METEOR score is calculated as the harmonic mean of precision and recall between the proposed and ground truth sentence given the set of alignments.

ROUGE (Lin, 2004) is another evaluation metric mainly used to evaluate the automatic summarization systems. There are five different extensions of ROUGE available: ROUGE-N, on N-gram based co-occurrence statistics; ROUGE-L, longest common subsequence (LCS) based statistics (Longest common subsequence problem takes into account sentence level structure similarity naturally and identifies longest co-occurring in sequence n-grams automatically.); ROUGE-W, weighted

LCS-based statistics that favors consecutive LCSes; ROUGE-S, skip-bigram based co-occurrence statistics (Skip-bigram is any pair of words in their sentence order.); and ROUGE-SU, skip-bigram plus unigram-based co-occurrence statistics. In text generation, ROUGE-L is the most commonly used metric in text generation tasks because the LCS is easy to measure the similarity between two sentences in the same order.

Embedding Based approaches consider the meaning of each word as defined by a word embedding, which assigns a vector to each word as opposed to the rest of the above metrics that consider n-gram matching scenarios. A word embedding learning method such as the one from Mikolov et al. (2013) is used to calculate these embeddings using distributional semantics; that is, they approximate the meaning of a word by considering how often it co-occurs with other words in the corpus. These embedding-based metrics usually approximate sentence-level embeddings using some heuristic to combine the vectors of the individual words in the sentence. The sentence-level embeddings between the generated and reference response are compared using a measure such as cosine distance.

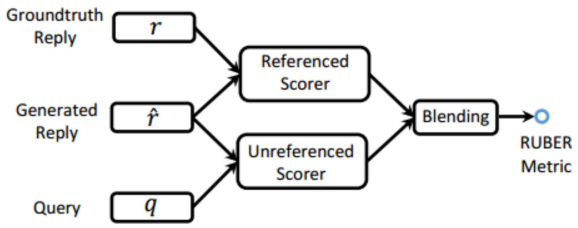

Fig. 3.9: Overview of RUBER metric. (Image borrowed from (Tao et al., 2017)

RUBER (Tao et al., 2017) is a Referenced metric and Unreferenced metric Blended Evaluation Routine for open-domain dialog systems. RUBER has the following distinct features: (*i*) An embedding-based scorer named *referenced metric*, which measures the similarity between a generated reply and the ground-truth. Instead of using word-overlapping information (as in BLEU and ROUGE), RUBER's reference metric measures the similarity by pooling of word embeddings (Forgues et al., 2014) which is more suited to dialog systems due to the diversity of replies. (*ii*) A neural network-based scorer named *unreferenced metric* that measures the relatedness between the generated reply and its query. This scorer is unreferenced because it does not refer to ground-truth and requires no manual annotation labels. (*iii*) The referenced and unreferenced metrics are combined with strategies like averaging further improves the performance (see Figure 3.9).

3.11 Summary

This chapter presents an extensive survey on current approaches in data-driven dialog modeling that use deep learning technologies, after some detailed introduction to various components of a spoken dialogue system including speech recognition, language understanding (spoken or text-based), dialogue manager, and language generation (spoken or text-based) . The chapter also describes available deep dialog modeling software and datasets suitable for research, development, and evaluation.

Deep learning technologies have yielded recent improvements in dialogue systems as well as new research activities. Most of the current dialogue systems and research on them are moving towards large-scale data-driven and specifically end-to-end trainable models. In addition to the current new approaches and datasets, also highlighted in this chapter are potential future directions in building conversational dialog systems including hierarchical structures, multi-agent systems as well as domain adaptation.

Dialogue systems, especially the spoken version, are a representative instance of multiple-stage information processing exemplified in NLP. The multiple stages include speech recognition, language understanding (Chapter 2), decision making (via dialogue manager), and language/speech generation. Such multiple-stage processing schemes suit ideally well deep learning methodology, which is based on end-to-end learning in multiple-layered (or deep) systems. The current progress in applying deep learning to dialogue systems as reviewed, in this chapter, has largely been limited to using deep learning to modeling and optimizing each individual processing stage in the overall system. The future progress is expected to broaden such a scope and to succeed in the fully end-to-end systems.

References

Asri, L. E., He, J., and Suleman, K. (2016). A sequence-to-sequence model for user simulation in spoken dialogue systems. *Interspeech*.

Aust, H., Oerder, M., Seide, F., and Steinbiss, V. (1995). The philips automatic train timetable information system. *Speech Communication*, 17:249–262.

Banchs, R. E. and Li., H. (2012). Iris:chat-oriented dialogue system based on the vector space model. *ACL*.

Banerjee, S. and Lavie, A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. *ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*.

Bapna, A., Tur, G., Hakkani-Tur, D., and Heck, L. (2017). Improving frame semantic parsing with hierarchical dialogue encoders.

Bateman, J. and Henschel, R. (1999). From full generation to near-templates without losing generality. In *KI'99 workshop, "May I Speak Freely?"*.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). Weight uncertainty in neural networks. *ICML*.

Bordes, A., Boureau, Y.-L., and Weston, J. (2017). Learning end-to-end goal-oriented dialog. In *ICLR 2017*.

Busemann, S. and Horacek, H. (1998). A flexible shallow approach to text generation. In *International Natural Language Generation Workshop, Niagara-on-the-Lake, Canada*.

Celikyilmaz, A., Sarikaya, R., Hakkani-Tur, D., Liu, X., Ramesh, N., and Tur, G. (2016). A new pre-training method for training deep learning models with application to spoken language understanding. In *Proceedings of Interspeech*, pages 3255–3259.

Chen, Y.-N., Hakkani-Tür, D., Tur, G., Gao, J., and Deng, L. (2016). End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In *Proceedings of The 17th Annual Meeting of the International Speech Communication Association (INTERSPEECH)*, San Francisco, CA. ISCA.

Crook, P. and Marin, A. (2017). Sequence to sequence modeling for user simulation in dialog systems. *Interspeech*.

Cuayahuitl, H., Yu, S., Williamson, A., and Carse, J. (2016). Deep reinforcement learning for multi-domain dialogue systems. *arXiv preprint arXiv:1611.08675*.

Cuayhuitl, H. (2016). Simpleds: A simple deep reinforcement learning dialogue system. *International Workshop on Spoken Dialogue Systems (IWSDS)*.

Dale, R. and Reiter, E. (2000). Building natural language generation systems. *Cambridge, U.K.: Cambridge University Press*.

Deng, L. (2016). Deep learning from speech recognition to language and multimodal processing. *APSIPA Transactions on Signal and Information Processing (Cambridge University Press)*.

Deng, L. and Li, X. (2013). Machine learning paradigms for speech recognition: An overview. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(5):1060–1089.

Deng, L. and Yu, D. (2015). Deep Learning: Methods and Applications, by NOW Publishers.

Dhingra, B., Li, L., Li, X., Gao, J., Chen, Y.-N., Ahmed, F., and Deng, L. (2016a). End-to-end reinforcement learning of dialogue agents for information access. *arXiv preprint arXiv:1609.00777*.

Dhingra, B., Li, L., Li, X., Gao, J., Chen, Y.-N., Ahmed, F., and Deng, L. (2016b). Towards end-to-end reinforcement learning of dialogue agents for information access. *ACL*.

Dodge, J., Gane, A., Zhang, X., Bordes, A., Chopra, S., Miller, A., Szlam, A., and Weston, J. (2015). Evaluating prerequisite qualities for learning end-to-end dialog systems. *arXiv preprint arXiv:1511.06931*.

Elhadad, M. and Robin, J. (1996). An overview of surge: A reusable comprehensive syntactic realization component. *Technical Report 96-03, Dept of Mathematics and Computer Science, Ben Gurion University, Beer Sheva, Israel*.

Fatemi, M., Asri, L. E., Schulz, H., He, J., and Suleman, K. (2016a). Policy networks with two-stage training for dialogue systems. *arXiv preprint arXiv:1606.03152*.

Fatemi, M., Asri, L. E., Schulz, H., He, J., and Suleman, K. (2016b). Policy networks with two-stage training for dialogue systems. *arXiv:1606.03152*.

Forgues, G., Pineau, J., Larcheveque, J.-M., and Tremblay, R. (2014). Bootstrapping dialog systems with word embeddings. *NIPS ML-NLP Workshop*.

Gasic, M., Jurcicek, F., Keizer, S., Mairesse, F., Thomson, B., Yu, K., and Young, S. (2010). Gaussian processes for fast policy optimisation of pomdp-based dialogue managers. *SIGDIAL*.

Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Wen, T.-H. (2015). Multi-agent learning in multi-domain spoken dialogue systems. *NIPS workshop on Spoken Language Understanding and Interaction*.

Gai, M., Mrki, N., Rojas-Barahona, L. M., Su, P.-H., Ultes, S., Vandyke, D., Wen, T.-H., and Young, S. (2016). Dialogue manager domain adaptation using gaussian process reinforcement learning. *Computer Speech and Language*.

Gai, M., Mrki, N., Su, P.-H., Vandyke, D., Wen, T.-H., and Young, S. (2015). Policy committee for adaptation in multi-domain spoken dialogue sytems. *ASRU*.

Ge, W. and Xu, B. (2016). Dialogue management based on multi-domain corpus. *Special Interest Group on Discourse and Dialog*.

Goller, C. and Kchler, A. (1996). Learning task-dependent distributed representations by backpropagation through structure. *IEEE*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *NIPS*.

Gorin, A. L., Riccardi, G., and Wright, J. H. (1997). How may i help you? *Speech Communication*, 23:113–127.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18:602–610.

Hakkani-Tür, D., Tur, G., Celikyilmaz, A., Chen, Y.-N., Gao, J., Deng, L., and Wang, Y.-Y. (2016). Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *Proceedings of Interspeech*, pages 715–719.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

He, J., Chen, J., He, X., Gao, J., Li, L., Deng, L., and Ostendorf, M. (2016). Deep reinforcement learning with a natural language action space. *ACL*.

He, X. and Deng, L. (2011). Speech recognition, machine translation, and speech translation a unified discriminative learning paradigm. In *IEEE Signal Processing Magazine*.

He, X. and Deng, L. (2013). Speech-centric information processing: An optimization-oriented approach. In *IEEE*.

Hemphill, C. T., Godfrey, J. J., and Doddington, G. R. (1990). The atis spoken language systems pilot corpus. *DARPA speech and natural language workshop*.

Henderson, M., Thomson, B., and Williams, J. D. (2014). The third dialog state tracking challenge. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 324–329. IEEE.

Henderson, M., Thomson, B., and Young, S. (2013). Deep neural network approach for the dialog state tracking challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 467–471.

Higashinaka, R., Imamura, K., Meguro, T., Miyazaki, C., Kobayashi, N., Sugiyama, H., Hirano, T., Makino, T., and Matsuo, Y. (2014). Towards an open-domain conversational system fully based on natural language processing. *COLING*.

Hinton, G., Deng, L., Yu, D., Dahl, G., rahman Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29(6):82–97.

Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A., and Heck, L. (2013). Learning deep structured semantic models for web search using click-through data. *ACM International Conference on Information and Knowledge Management (CIKM)*.

Huang, X. and Deng, L. (2010). An overview of modern speech recognition. In *Handbook of Natural Language Processing, Second Edition, Chapter 15*.

Jaech, A., Heck, L., and Ostendorf, M. (2016). Domain adaptation of recurrent neural networks for natural language understanding.

K., G., J., H., and O, L. (2005). Learning user simulations for information state update dialogue systems. *9th European Conference on Speech Communication and Technology (INTERSPEECH - EUROSPEECH)*.

K., G., J., H., and O, L. (2006). User simulation for spoken dialogue systems: Learning and evaluation. *INTERSPEECH - EUROSPEECH*.

Kannan, A. and Vinyals, O. (2016). Adversarial evaluation of dialog models. *Workshop on Adversarial Training, NIPS 2016, Barcelona, Spain*.

Kim, Y.-B., Stratos, K., and Kim, D. (2017a). Adversarial adaptation of synthetic or stale data. *ACL*.

Kim, Y.-B., Stratos, K., and Kim, D. (2017b). Domain attention with an ensemble of experts. *ACL*.

Kim, Y.-B., Stratos, K., and Sarikaya, R. (2016a). Domainless adaptation by constrained decoding on a schema lattice. *COLING*.

Kim, Y.-B., Stratos, K., and Sarikaya, R. (2016b). Frustratingly easy neural domain adaptation. *COLING*.

Kumar, A., Irsoy, O., Su, J., Bradbury, J., English, R., Pierce, B., Ondruska, P., Gulrajani, I., and Socher, R. (2015). Ask me anything: Dynamic memory networks for natural language processing. *Neural Information Processing Systems (NIPS)*.

Kurata, G., Xiang, B., Zhou, B., and Yu, M. (2016). Leveraging sentence level information with encoder lstm for natural language understanding. *arXiv preprint, arXiv:1601.01530*.

Langkilde, I. and Knight, K. (1998). Generation that exploits corpus-based statistical knowledge. *ACL*.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *IEEE*, 86:2278–2324.

Lemon, O. and Rieserr, V. (2009). Reinforcement learning for adaptive dialogue systems - tutorial. *EACL*.

Li, J., Deng, L., Gong, Y., and Haeb-Umbach, R. (2014). An overview of noise-robust automatic speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(4):745–777.

Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, B. (2016a). A diversity-promoting objective function for neural conversation models. *NAACL*.

Li, J., Galley, M., Brockett, C., Spithourakis, G. P., Gao, J., and Dolan, B. (2016b). A persona based neural conversational model. *ACL*.

Li, J., Monroe, W., Shu, T., Jean, S., Ritter, A., and Jurafsky, D. (2017). Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*.

Li, L., Balakrishnan, S., and Williams, J. (2009). Reinforcement learning for dialog management using least-squares policy iteration and fast feature selection. *InterSpeech*.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. *In Text summarization branches out: ACL-04 Workshop*.

Lipton, Z. C., Li, X., Gao, J., Li, L., Ahmed, F., and Deng, L. (2016). Efficient dialogue policy learning with bbq-networks. *arXiv.org*.

Lison, P. (2013). *Structured Probabilistic Modelling for Dialogue Management*. Department of Informatics Faculty of Mathematics and Natural Sciences University of Osloe.

Liu, B. and Lane, I. (2016a). Attention-based recurrent neural network models for joint intent detection and slot filling. *Interspeech*.

Liu, B. and Lane, I. (2016b). Attention-based recurrent neural network models for joint intent detection and slot filling. In *SigDial*.

Liu, C.-W., Lowe, R., Serban, I. V., Noseworthy, M., Charlin, L., and Pineau, J. (2016). How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *EMNLP*.

Lowe, R., Pow, N., Serban, I. V., Charlin, L., and Pineau, J. (2015a). Incorporating unstructured textual knowledge sources into neural dialogue systems. *Neural Information Processing Systems Workshop on Machine Learning for Spoken Language Understanding*.

Lowe, R., Pow, N., Serban, I. V., and Pineau, J. (2015b). The ubuntu dialogue corpus: A large dataset for research in unstructure multi-turn dialogue systems. In *SIGDIAL 2015*.

Mairesse, F. and Walker, M. A. (2011). Controlling user perceptions of linguistic style: Trainable generation of personality traits. *Computer Linguistics*.

Mairesse, F. and Young, S. (2014). Stochastic language generation in dialogue using factored language models. *Computer Linguistics*.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al. (2015). Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):530–539.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mizil, C. D. N. and Lee, L. (2011). Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *NIPS Deep Learning Workshop*.

Mrkšić, N., Séaghdha, D. Ó., Wen, T.-H., Thomson, B., and Young, S. (2016). Neural belief tracker: Data-driven dialogue state tracking. *arXiv preprint arXiv:1606.03777*.

Oh, A. H. and Rudnicky, A. I. (2000). Stochastic language generation for spoken dialogue systems. *ANLP/NAACL Workshop on Conversational Systems*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). Bleu: a method for automatic evaluation of machine translation. *40th annual meeting on Association for Computational Linguistics (ACL)*.

Passonneau, R. J., Epstein, S. L., Ligorio, T., and Gordon, J. (2011). Embedded wizardry. *SIGDIAL 2011 Conference*.

Peng, B., Li, X., Li, L., Gao, J., Celikyilmaz, A., Lee, S., and Wong, K.-F. (2017). Composite task-completion dialogue system via hierarchical deep reinforcement learning. *arxiv:1704.03084v2*.

Pietquin, O., Geist, M., and Chandramohan, S. (2011a). Sample efficient on-line learning of optimal dialogue policies with kalman temporal differences. *IJCAI 2011, Barcelona, Spain*.

Pietquin, O., Geist, M., Chandramohan, S., and FrezzaBuet, H. (2011b). Sample-efficient batch reinforcement learning for dialogue management optimization. *ACM Transactions on Speech and Language Processing*.

Ravuri, S. and Stolcke, A. (2015). Recurrent neural network and lstm models for lexical utterance classification. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Ritter, A., Cherry, C., and Dolan., W. B. (2011). Data-driven response generation in social media. *Empirical Methods in Natural Language Processing*.

Sarikaya, R., Hinton, G. E., and Deoras, A. (2014). Application of deep belief networks for natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(4):778–784.

Sarikaya, R., Hinton, G. E., and Ramabhadran, B. (2011). Deep belief nets for natural language call-routing. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5680–5683. IEEE.

Schatzmann, J., Weilhammer, K., and Matt Stutle, S. Y. (2006). A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The Knowledge Engineering Review*.

Serban, I., Klinger, T., Tesauro, G., Talamadupula, K., Zhou, B., Bengio, Y., and Courville, A. (2016a). Multiresolution recurrent neural networks:an application to dialogue response generation. *arXiv preprint arXiv:1606.00776v2*.

Serban, I., Sordoni, A., and Bengio, Y. (2017). A hierarchical latent variable encoder-decoder model for generating dialogues. *AAAI*.

Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. (2015). Building end-to-end dialogue systems using generative hierarchical neural network models. *AAAI*.

Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. (2016b). Building end-to-end dialogue systems using generative hierarchical neural networks. *AAAI*.

Shah, P., Hakkani-Tur, D., and Heck, L. (2016). Interactive reinforcement learning for task-oriented dialogue management. *SIGDIAL*.

Shang, L., Lu, Z., and Li, H. (2015). Neural responding machine for short text conversation. *ACL-IJCNLP*.

Simonnet, E., Camelin, N., Deléglise, P., and Estève, Y. (2015). Exploring the use of attention-based recurrent neural networks for spoken language understanding. In *Machine Learning for Spoken Language Understanding and Interaction NIPS 2015 workshop (SLUNIPS 2015)*.

Simpson, A. and Eraser, N. M. (1993). Black box and glass box evaluation of the sundial system. *Third European Conference on Speech Communication and Technology*.

Singh, S. P., Kearns, M. J., Litman, D. J., and Walker, M. A. (2016). Reinforcement learning for spoken dialogue systems. *NIPS*.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J., Gao, J., and Dolan, B. (2015a). A neural network approach to context-sensitive generation of conversational responses. *North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2015)*.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015b). A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205, Denver, Colorado. Association for Computational Linguistics.

Stent, A. (1999). Content planning and generation in continuous-speech spoken dialog systems. In *KI'99 workshop, "May I Speak Freely?"*.

Stent, A., Prasad, R., and Walker, M. (2004). Trainable sentence planning for complex information presentation in spoken dialog systems. *ACL*.

Su, P.-H., Gasic, M., Mrksic, N., Rojas-Barahona, L., Ultes, S., Vandyke, D., Wen, T.-H., and Young, S. (2016). On-line active reward learning for policy optimisation in spoken dialogue systems. *arXiv preprint arXiv:1605.07669*.

Sukhbaatar, S., Weston, J., Fergus, R., et al. (2015). End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Sutton, R. S. and Singh, S. P. (1999). Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112:181–211.

Tafforeau, J., Bechet, F., Artières, T., and Favre, B. (2016). Joint syntactic and semantic analysis with a multitask deep learning framework for spoken language understanding. In *Interspeech*, pages 3260–3264.

Tao, C., Mou, L., Zhao, D., and Yan, R. (2017). Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems. *ArXiv2017*.

Thomson, B. and Young, S. (2010). Bayesian update of dialogue state: A pomdp framework for spoken dialogue systems. *Computer Speech and Language*, 24(4):562–588.

Tur, G. and De Mori, R. (2011). *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Tur, G. and Deng, L. (2011). *Intent Determination and Spoken Utterance Classification, Chapter 4 in Book: Spoken Language Understanding*. John Wiley and Sons, New York, NY.

Tur, G., Deng, L., Hakkani-Tür, D., and He, X. (2012). Towards deeper understanding: Deep convex networks for semantic utterance classification. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5045–5048. IEEE.

Vinyals, O. and Le, Q. (2015). A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Walker, M., Stent, A., Mairesse, F., and Prasad, R. (2007). Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research*.

Wang, Z., Stylianou, Y., Wen, T.-H., Su, P.-H., and Young, S. (2015). Learning domain-independent dialogue policies via ontology parameterisation. In *SIGDAIL*.

Wen, T.-H., Gasic, M., Mrksic, N., Rojas-Barahona, L. M., hao Su, P., Ultes, S., Vandyke, D., and Young, S. J. (2016a). A network-based end-to-end trainable task-oriented dialogue system. *arXiv*.

Wen, T.-H., Gasic, M., Mrksic, N., Rojas-Barahona, L. M., Su, P.-H., Ultes, S., Vandyke, D., and Young, S. (2016b). A network-based end-to-end trainable task-oriented dialogue system. *arXiv preprint arXiv:1604.04562*.

Wen, T.-H., Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S. (2015a). Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *EMNLP*.

Wen, T.-H., Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S. (2015b). Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

Weston, J., Chopra, S., and Bordesa, A. (2015). Memory networks. In *International Conference on Learning Representations (ICLR)*.

Williams, J., Raux, A., and Handerson, M. (2016). The dialog state tracking challenge series: A review. *Dialogue and Discourse*, 7(3):4–33.

Williams, J. D., Raux, A., Ramachandran, D., and Black, A. W. (2013). The dialog state tracking challenge. In *SIGDIAL Conference*, pages 404–413.

Williams, J. D. and Zweig, G. (2016a). End-to-end lstm-based dialog control optimized with supervised and reinforcement learning. *arXiv preprint arXiv:1606.01269*.

Williams, J. D. and Zweig, G. (2016b). End-to-end lstm-based dialog control optimized with supervised and reinforcement learning. *arXiv*.

Xu, P. and Sarikaya, R. (2013). Convolutional neural network based triangular CRF for joint intent detection and slot filling. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 78–83. IEEE.

Yao, K., Zweig, G., Hwang, M.-Y., Shi, Y., and Yu, D. (2013). Recurrent neural networks for language understanding. In *INTERSPEECH*, pages 2524–2528.

Young, S. (2015). Open domain statistical spoken dialogue systems. *SLSP Presentation Slides http://grammars.grlmc.com/SLSP2015/Slides/d3s1ODSDS.pdf*.

Yu, D. and Deng, L. (2015). Automatic speech recognition a deep learning approach. *Springer*.

Yu, Y., Eshghi, A., and Lemon, O. (2016). Training an adaptive dialogue policy for interactive learning of visually grounded word meanings. *SIGDIAL*.

Yu, Z., Black, A., and Rudnicky, A. I. (2017). Learning conversational systems that interleave task and non-task content. *arXiv:1703.00099v1*.

Yu, Z., Papangelis, A., and Rudnicky, A. (2015). Ticktock: A non-goal-oriented multimodal dialog system with engagement awareness. *AAAI Spring Symposium*.

Li Deng · Yang Liu

Deep Learning in Natural Language Processing

In recent years, deep learning has fundamentally changed the landscapes of a number of areas in artificial intelligence, including speech, vision, natural language, robotics, and game playing. In particular, the striking success of deep learning in a wide variety of natural language processing (NLP) applications has served as a benchmark for the advances in one of the most important tasks in artificial intelligence.

This book reviews the state of the art of deep learning research and its successful applications to major NLP tasks, including speech recognition and understanding, dialogue systems, lexical analysis, parsing, knowledge graphs, machine translation, question answering, sentiment analysis, social computing, and natural language generation from images. Outlining and analyzing various research frontiers of NLP in the deep learning era, it features self-contained, comprehensive chapters written by leading researchers in the field. A glossary of technical terms and commonly used acronyms in the intersection of deep learning and NLP is also provided.

The book appeals to advanced undergraduate and graduate students, post-doctoral researchers, lecturers and industrial researchers, as well as anyone interested in deep learning and natural language processing.

The book offers a unique reference guide for practitioners in various sectors, especially the Internet and AI start-ups, where NLP technologies are becoming an essential enabler and a core differentiator. **Hongjiang Zhang (Founder, Sourcecode Capital; former CEO of KingSoft)**

The book is clearly structured, and moves from major research trends, to the latest deep learning approaches, to their limitations and promising future work. **Haifeng Wang (Executive Vice President, Baidu; former President of ACL)**

I highly recommend that speech and NLP researchers, engineers, and students read this outstanding and timely book, not only to learn about the state of the art in NLP and deep learning, but also to gain vital insights into what the future of the NLP field will hold. **Sadaoki Furui (President, Toyota Technological Institute at Chicago)**

There is a renewed sense of excitement with the rise of deep learning, especially in speech and vision, and perhaps in language. This book gives a sense of where the excitement in language is coming from. **Ken Church (IBM research; former President of ACL)**

Computer Science

ISBN 978-981-10-5208-8

▶ springer.com

Deng · Liu

Deep Learning in Natural Language Processing