Cross-Lingual Approaches to Reference Resolution in Dialogue Systems

Amr Sharaf[1], **Arpit Gupta**[2], **Hancheng Ge**[2], **Chetan Naik**[2] **and Lambert Mathias**[2]

[1]University of Maryland
[2]Amazon Alexa AI
[1]amr@cs.umd.edu
[2] {arpgup,ghanche,chetnaik,mathiasl}@amazon.com

Abstract

In the slot-filling paradigm, where a user can refer back to slots in the context during the conversation, the goal of the contextual understanding system is to resolve the referring expressions to the appropriate slots in the context. In this paper, we build on (Naik et al., 2018), which provides a scalable multi-domain framework for resolving references. However, scaling this approach across languages is not a trivial task, due to the large demand on acquisition of annotated data in the target language. Our main focus is on cross-lingual methods for reference resolution as a way to alleviate the need for annotated data in the target language. In the cross-lingual setup, we assume there is access to annotated resources as well as a well trained model in the source language and little to no annotated data in the target language. In this paper, we explore three different approaches for cross-lingual transfer — delexicalization as data augmentation, multilingual embeddings and machine translation. We compare these approaches both on a low resource setting as well as a large resource setting. Our experiments show that multilingual embeddings and delexicalization via data augmentation have a significant impact in the low resource setting, but the gains diminish as the amount of available data in the target language increases. Furthermore, when combined with machine translation we can get performance very close to actual live data in the target language, with only 25% of the data projected into the target language.

1 Introduction

Most commercial spoken language systems consist of multiple components (Tur and De Mori, 2011). Figure 1 describes this architecture - a user query is presented as text (this could be the output of a speech recognizer in case of spoken language input) to a collection of domain-specific natural language understanding (NLU) systems, which produces the most likely semantic interpretation - typically represented as intents and slots (Wang et al., 2011). The resulting semantic interpretation is then passed to a reference resolver, in order to resolve referring expressions (including anaphora) to their antecedent slots in the conversation, which is then sent to the dialogue manager whose main responsibility is to determine the next action. This action is the input to the natural language generation component that is responsible for generating the system response back to the user.

In this paper, we focus on the reference resolution task. Resolving anaphora and referring expressions is an important sub-component and is essential for maintaining the state of the conversation across turns (Celikyilmaz et al., 2014). The key here is to leverage the dialogue context effectively (Bhargava et al., 2013; Xu and Sarikaya, 2014) for improving spoken language understanding accuracy. However, in commercial systems like Siri, Google Home and Alexa, the NLU component is a highly federated

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

and diverse collection of services spanning rules or diverse statistical models. Typical end-to-end approaches Bapna et al. (2017) which require back-propagation through the NLU sub-systems are not feasible in such a setting, effectively the NLU systems could be considered immutable. This issue is addressed in (Naik et al., 2018), which describes a scalable multi-domain architecture for solving the reference resolution problem called *context carryover*, and does not require coupling with the multiple NLU sub-systems. However, the focus of that work is mainly on the monolingual setting where annotated resources are readily available. In this paper, our main contribution is to show how *context carryover* can be extended to new languages effectively using cross-lingual transfer.

While a simple naive approach would be to collect annotated resources in the target language, this is often expensive and time consuming. In the cross-lingual setting, we assume we have access to annotated data in the source language (en_US) on which we can train the context carryover model. For the target language (de_DE), we assume we have little to no annotated data available, which is typically the case in a low-resource setting. In this paper, we empirically investigate three approaches for cross-lingual transfer, multilingual embeddings, translation projection and delexicalization. We show that translation is an effective strategy to bootstrap a model in the target domain. Furthermore, we demonstrate that data augmentation strategies that abstract from the lexical representation can significantly boost the performance of the models, especially in the low resource setting.

This paper is structured as follows. In Section 2.1, we give a formal description of the carryover task. In Section 2.2, we outline multiple approaches to cross-lingual transfer. In Section 3, we present empirical results comparing and analyzing the effectiveness of multiple strategies for cross-lingual training of these models. Finally, we present our conclusions and a few possible areas of further research.

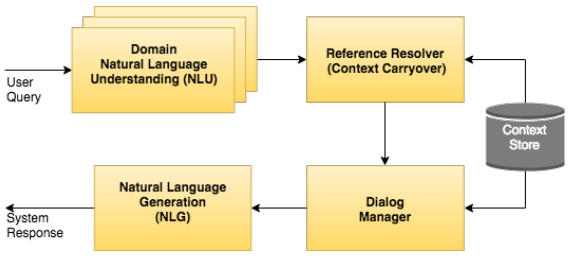

Figure 1: Spoken Dialogue Architecture: A pipelined approach for spoken language understanding. The context carryover system is responsible for resolving references in a conversation and is an input to the dialogue manager.

2 Approach

2.1 Task Definition

We first motivate the context carryover task with the example shown in Table 1. At turn **U3**, the context carryover system resolves the implicit reference by determining that the contextual slots (*Place=Exploratorium*) and (*City=san francisco*) are most relevant to the user utterance "*what's the address*", but (*PlaceType=museum*) is no longer relevant. During training the model learns the carryover action by leveraging the similarities between the candidate slot and the dialogue context. At test time, the context carryover system is activated only on a user turn for which dialog context is available, and the decision for each slot is made independently.

We can now formally define the context carryover task. We define a dialogue turn at time t as the tuple $\{a_t, \boldsymbol{S}_t, \boldsymbol{w}_t\}$, where $\boldsymbol{w_t} \in \mathcal{W}$ is a sequence of words $\{w_{it}\}_{i=1}^{N_t}$; $a_t \in \mathcal{A}$ is the dialogue act; and $\boldsymbol{S}_t$ is a set of slots, where each slot s is a key value pair $s = \{k, v\}$, with $k \in \mathcal{K}$ being the slot name (or slot key), and $v \in \mathcal{V}$ being the slot value. $\boldsymbol{u}_t = \{a_t^u, \boldsymbol{S}_t^u, \boldsymbol{w}_t^u\}$ represents a user-initiated turn and $\boldsymbol{v}_t = \{a_t^v, \boldsymbol{S}_t^v, \boldsymbol{w}_t^v\}$ represents a system initiated turn. Given a sequence of D user turns $\{\boldsymbol{u}_{t-D}, \ldots, \boldsymbol{u}_{t-2}, \boldsymbol{u}_{t-1}\}$; and their associated system turns $\{\boldsymbol{v}_{t-D}, \ldots, \boldsymbol{v}_{t-2}, \boldsymbol{v}_{t-1}\}$[1]; and the

[1]For simplicity we assume a turn taking model - a user turn and system turn alternate.

Domain	Turns	Current Turn Slots	Carryover Slot
Local	**U1**: find a museum in san francisco	PlaceType=museum City=san francisco	-
Local	**V1**: Found exploratorium it is 10 miles away	Place=Exploratorium Distance=10 miles	
Local	**U2**: what's the address ***<Implicit Reference>***		Place=Exploratorium City=san francisco
Local	**V2**: located on Embarcadero st..	Location=Pier 15, Embarcadero St	
Calling	**U3**: call them ***<Explicit Reference>***		Contact=Exploratorium

Table 1: Context Carryover resolves explicit and implicit references in the dialogue - at each turn the most relevant slot from the context is 'carried over' to the current turn. The conversation from U2 to U3 involves a domain change - Local to Calling - and a schema change i.e the slot Place from the Local domain needs to be carried over and transformed to Contact in the Calling domain.

current user turn $\boldsymbol{u}_t$, we construct a candidate set of slots from the context

$$C(\boldsymbol{S}) = \bigcup_{i \in u,v, j=t-D}^{t-1} \boldsymbol{S}_j^i \tag{1}$$

For a candidate slot $s \in C(\boldsymbol{S})$, for the dialogue turn at time t, we formulate context carryover as a binary classification task

$$P(+1|s, \boldsymbol{u}_t; \boldsymbol{u}_{t-D}^{t-1}, \boldsymbol{v}_{t-D}^{t-1}; d(s)) > \tau \tag{2}$$

where, τ is a tunable decision threshold. $d(s) \in [0, D]$ is an integer value describing the offset of the slot from the current turn $\boldsymbol{u}_t$. We use a neural network encoder-decoder formulation to model this task as shown in Figure 2. For a detailed description of the model, the reader is referred to (Naik et al., 2018).

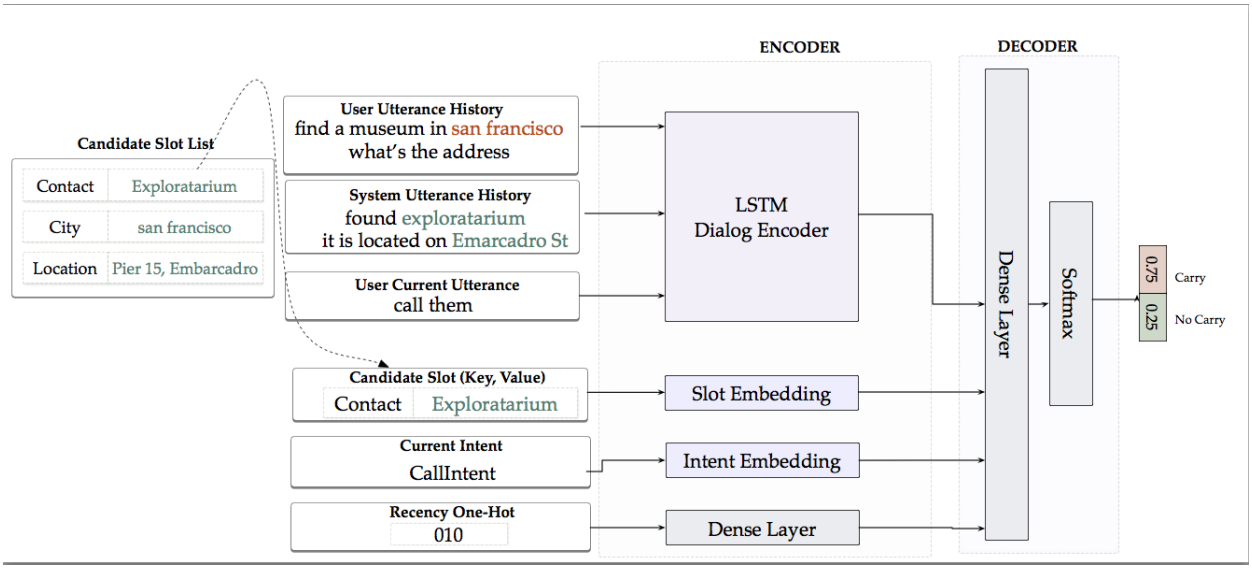

Figure 2: Context Carryover Architecture: Resolving slot references across a collection of diverse NLU sub-systems.

2.2 Cross-Lingual Model Transfer

In order to alleviate the cost of obtaining annotated resources in a new language, we focus on cross-lingual techniques. In the cross-lingual setting, a model is trained on a source language with sufficiently large annotated data, and the learnt information is then transferred to a target language with low resources or even zero training data (Pan et al., 2010; Kozhevnikov and Titov, 2013). In this work, we compare and contrast 3 approaches to cross-lingual transfer.

2.2.1 Multilingual word embeddings

We initialize the models with multilingual word embeddings where the words in various languages have been projected into a common language space. Here, we leverage the fasttext embeddings as described in (Lample et al., 2018). The hypothesis is that because the words across languages that share the same semantics are in the same vector space, training on the source language should result in effective transfer to the target language without having to rely on a lot of target language data.

2.2.2 Translation projection

Similar to the work done in (Lefevre et al., 2010), we use a statistical machine translation system to translate the source language into the target language, and then we train the model in the translated target language. Although the training data is likely to include errors due to translation inaccuracy, this method can potentially predict the correct semantics by learning from consistent error patterns.

2.2.3 Delexicalization

We follow the approach outlined in (Henderson et al., 2014) where we replace the input streams to the model - the context turns, the current turn and the slots - with generic symbols representing the slot key and the intent patterns associated with those turns, as shown in 2. We augment the *original* training data with this delexicalized data. This allows the model to generalize to unseen lexical tokens, by focusing the model via the attention mechanism on the semantic representation.

Turns	Intent	Slots	Delexicalized Turn
U1: find a museum in san francisco	Local.SearchPlaceIntent	PlaceType=museum City=san francisco	Local.SearchPlaceIntent find a PlaceType in City
V1: Found exploratorium it is 10 miles away	Local.InformAction	Place=Exploratorium Distance=10 miles	Local.InformAction found Place it is Distance away.

Table 2: The user and system utterance tokens are delexicalized by replacing the tokens with the slot keys and inserting the intent label at the beginning.

3 Experiments

3.1 Data Setup

For the experiments, we present our results on a sampled internal benchmark dataset for voice based applications that spans six domains — weather, music, video, local business search, movie showtimes and general question answering. Table 3 summarizes the statistics in the training, development and test sets across different domains. We focus on en_US and de_DE as the two languages for all our multilingual experiments. Both datasets have very similar distribution of average number of user turns.

	en_US			de_DE		
	Train	Dev	Test	Train	Dev	Test
Number of Sessions	18k	15k	15k	17k	13k	12k
Average turns per session	2.2	2.14	2.18	2.20	2.0	2.17

Table 3: Context Carryover Data Setup

3.1.1 Training Setup and Evaluation Metrics

For the model, we initialize the word embeddings using fasttext embedings (Lample et al., 2018). The model is trained using mini-batch SGD with Adam optimizer (Kinga and Adam, 2015) with standard parameters to minimize the class weighted cross-entropy loss. In our experiments, we use 128 dimensions for the LSTM hidden states and 256 dimensions for the hidden state in the decoder. Similar to (Wiseman et al., 2015), we pre-train an LSTM encoder using live data in each of the languages and use this model to initialize the parameters of the LSTM-based encoders. All model setups are trained for 40 epochs. For evaluation on test set we pick the best model based on performance on dev set. For evaluation, we only select those slots as the final hypothesis, whose $\tau > 0.5$ and occur in the context of the conversation. For each utterance, independent carryover decisions are taken for each candidate slot. We use standard definitions of precision, recall and F1 by comparing the reference slots with the model hypothesis slots. If an entity type is repeated in the current turn then we do not carry this from dialogue history.

3.2 Establishing the monolingual baselines

For comparison, we introduce a simple naive baseline that carries over the most recent slots in the dialogue session, that demonstrates the complexity of the task - simply carrying over the slot from the context results in very poor performance in the context carryover task. Table 4 establishes the monolingual baseline for en_US and de_DE based on available annotated data described in Table 3, and using the neural network architecture described earlier.

	en_US			de_DE		
Method	Precision	Recall	F1	Precision	Recall	F1
Baseline	37.77	93.75	53.85	30.57	85.66	45.06
Encoder-Decoder w/ attention	90.7	93.1	91.9	77.6	69.9	73.07

Table 4: Monolingual Baseline Experiments for en_US and de_DE. The train and test sets are on fully annotated live datasets. For de_DE this represents the best performance when annotated data is available in the target domain.

3.3 Cross-lingual Transfer Experiment Results

Training a contextual slot carryover model across each locale independently is challenging, primarily due to the cost of acquiring labeled conversational data. Instead, we focus on methods for leveraging cross-lingual information. For all the experiments, we tune the model on the de_DE development set, and evaluate the results on the de_DE test. This dev and test data composition is the one described in Table 3.

3.3.1 Impact of multilingual embeddings

Here we compare the monolingually trained fasttext embeddings (Bojanowski et al., 2017) to the multilingually aligned embeddings (Conneau et al., 2018). In order to take advantage of the multilingual embeddings we train the model jointly on en_US data and en_US data translated to de_DE data, and evaluate the models on true de_DE test data. We find that multilingual embeddings help in the low resource setting, but as the available training data increases, the gains diminish [2].

3.3.2 Impact of Translation Projection

Here we use an in-house en_US to de_DE translation system. We translate all en_US data described in Table 3. In order to estimate the translation quality, we perform back translation from de_DE to en_US; the resulting translation when compared to the reference input en_US utterances gives us a BLEU (Papineni et al., 2002) score of 29.36.

From Table 6, we can see that as we increase the amount of translated data, the performance of the models on the target language increases significantly. While the performance on translation data is

[2] In our setup, we also found that multilingual embeddings only help when doing multilingual training i.e jointly training across en_US and de_DE

% translated en_US->de_DE	Embedding	Precision	Recall	F1
1%	Monolingual	63.9	60.0	61.6
	Multilingual	62.5	64.2	**63.3**
25%	Monolingual	72.5	59.6	63.1
	Multilingual	73.5	62.8	**66.4**
100%	Monolingual	71.2	63.1	66.0
	Multilingual	72.9	62.5	65.9

Table 5: Impact of multilingual embeddings on test de_DE data: Multilingual embeddings provide diminishing gains as the amount of de_DE training data increases.

% translated en_US->de_DE	Precision	Recall	F1
1%	50.76	51.5	50.44
25%	70.5	60.4	63.6
100%	72.1	63.8	66.8

Table 6: Impact of translation projection on live de_DE test data for varying amounts of translated en_US->de_DE data. Compared to the monolingual de_DE baseline in Table 4, translation projection is within 10% relative F1, indicating that the noise in translation results in poorer performance.

still within 10% relative F1 compared to a model trained on live de_DE data (as shown in Table 4), this shows that translation projection provides a viable alternative to bootstrap context carryover models in a new language.

3.3.3 Impact of Delexicalization

In many cases, if we have access to the source language data, we can leverage it in addition to training on the synthesized translated data in the target language. In this paper, we leverage the available source language en_US data in two ways. We first consider the impact of delexicalization, described in Section 2.2.3. In this setting, we simply augment the target language de_DE data with delexicalized data from en_US. Furthermore, we also study the impact of initializing the model from trained en_US parameters (we call this source language initialization) vs training from scratch. Table 7, describes in detail the various experiments. We see that for the low resource setting, both delexicalization and source language initialization of the models have a huge impact on the performance on the target language. We get a 26% relative F1 improvement by leveraging these techniques. However, the gains diminish as we increase the amount of target language data. While delexicalization still gives a $4-9\%$ lift in relative F1 scores, we no longer get any gains from initializing the model from en_US data. Also, we see that we can get most of the gains from only translating 25% of the source language data, which indicates that we can bootstrap a new language with far fewer resources in the target language.

% translated en_US->de_DE	Delexicalization	Source Language initialization	Precision	Recall	F1
1%	No	No	50.76	51.5	50.44
	Yes	No	63.7	57.0	59.0
	Yes	Yes	64.2	62.9	**63.6**
25%	No	No	70.5	60.4	63.6
	Yes	No	68.7	70.0	**69.3**
	Yes	Yes	68.9	67.8	68.3
100%	No	No	72.1	63.8	66.8
	Yes	No	68.2	71.8	**69.8**
	Yes	Yes	71.5	66.7	68.8

Table 7: Impact of translation projection on live de_DE test data for varying amounts of translated en_US->de_DE training data. With only 25% of the translated data, combining with delexicalization we get significant improvements, within 4% F1 of the monolingual de_DE system. Source language initialization does not help except in the low resource setting when the target language data is in the 1% range.

4 Related Work

Most of the approaches for contextual understanding and dialogue have focused on en_US as the language, due to a wide variety of available resources (Bapna et al., 2017; Chen et al., 2016; Henderson et al., 2013). However, there is very little focus on multilingual contextual understanding. Recently, DTSC5 introduced a challenge where data was provided for training and developing in en_US, and the systems were evaluated on Chinese (Williams et al., 2013). (Hori et al., 2016) describes a combined en_US and rule based system to solve the task. However, this approach relies more on language agnostic rules, and does not really exploit any other data source in the target language.

Cross-lingual model transfer consists of modifying a source language model to make it directly applicable to a new language. This usually involves constructing a shared feature representation across the two languages. (McDonald et al., 2011) successfully apply this idea to the transfer of dependency parsers, using part-of-speech tags as the shared representation of words. A later extension of (Täckström et al., 2012) enriches this representation with cross-lingual word clusters, considerably improving the performance. (Kozhevnikov and Titov, 2013) used cross-lingual model transfer to learn a model for Semantic Role Labeling (SRL). The model combines both syntactic representations shared across different languages (like universal part-of-speech tags) as well as semantic shared representations using cross-lingual word clusters and cross- lingual distributed word representations. Closely related to our work (Lefevre et al., 2010) use a machine translation based approach to solve the data sparsity in the target language. for spoken language understanding systems.

While the above work has been leveraged for multiple tasks, there is no single comparison of the efficacy of these various strategies for contextual understanding and specifically reference resolution tasks. In this paper, we provide detailed experiments contrasting and comparing these different approaches for cross-lingual transfer and clearly demonstrate the effectiveness of this approach in both low-resource and large-resource settings.

5 Conclusion

In this work, we presented a cross-lingual extension of the *context carryover task* for contextual interpretation of slots in a multi-domain large-scale dialogue system. We explored three different approaches for cross-lingual transfer — multilingual embeddings, translation projection and delexicalization. Our empirical results on en_US and de_DE demonstrate that translation is an effective way to bootstrap systems in a new language. For the context carryover task, specifically, when combining translation with delexicalization, we only need 25% of the translated data to get within 4% F1 score of a system trained on true de_DE annotated collections. We also showed that multilingual embeddings give a small boost in the low data regime, but are not really useful when there is sufficient data, even noisy translated data, in the target domain.

In the future, we plan to improve our model by leveraging sub-word information which is crucial for robustness to rich morphology in some languages. We also plan on analyzing asian languages like Japanese and Chinese, to see if the above approach generalizes to languages with very different character distributions. Finally, our goal is to reduce reliance on an existence of a translation system, and exploring multi-task objectives where we can learn both language specific and task specific parameters within the same network architecture.

References

Ankur Bapna, Gokhan Tur, Dilek Hakkani-Tur, and Larry Heck. 2017. Sequential dialogue context modeling for spoken language understanding. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 103–114.

A. Bhargava, Asli Çelikyilmaz, Dilek Z. Hakkani-Tür, and Ruhi Sarikaya. 2013. Easy contextual intent prediction and slot detection. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8337–8341.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *TACL*, 5:135–146.

Asli Celikyilmaz, Zhaleh Feizollahi, Dilek Hakkani-Tur, and Ruhi Sarikaya. 2014. Resolving referring expressions in conversational dialogs for natural user interfaces. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2094–2104.

Yun-Nung Chen, Dilek Hakkani-Tür, Gökhan Tür, Jianfeng Gao, and Li Deng. 2016. End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In *INTERSPEECH*, pages 3245–3249.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. *ICLR*.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 360–365. IEEE.

Matthew Henderson, Blaise Thomson, and Steve J. Young. 2013. Deep neural network approach for the dialog state tracking challenge. In *SIGDIAL Conference*.

Takaaki Hori, Hai Wang, Chiori Hori, Shinji Watanabe, Bret Harsham, Jonathan Le Roux, John R Hershey, Yusuke Koji, Yi Jing, Zhaocheng Zhu, et al. 2016. Dialog state tracking with attention-based sequence-to-sequence learning. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pages 552–558. IEEE.

D Kinga and J Ba Adam. 2015. A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, volume 5.

Mikhail Kozhevnikov and Ivan Titov. 2013. Cross-lingual transfer of semantic role labeling models. *ACL*.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. *ICLR*.

Fabrice Lefevre, François Mairesse, and Steve Young. 2010. Cross-lingual spoken language understanding from unaligned data using discriminative classification models and machine translation. In *Eleventh Annual Conference of the International Speech Communication Association*.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the conference on empirical methods in natural language processing*, pages 62–72. Association for Computational Linguistics.

Chetan Naik, Arpit Gupta, Hancheng Ge, Lambert Mathias, and Ruhi Sarikaya. 2018. Contextual slot carryover for disparate schemas. In *Interspeech*.

Sinno Jialin Pan, Qiang Yang, et al. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. 2012. Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 477–487. Association for Computational Linguistics.

Gokhan Tur and Renato De Mori. 2011. *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Ye-Yi Wang, Li Deng, and Alex Acero. 2011. Semantic frame-based spoken language understanding. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, pages 41–91.

Jason Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. 2013. The dialog state tracking challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 404–413.

Sam Wiseman, Alexander M. Rush, Stuart M. Shieber, and Jason Weston. 2015. Learning anaphoricity and antecedent ranking features for coreference resolution. In *ACL*.

Puyang Xu and Ruhi Sarikaya. 2014. Contextual domain classification in spoken language understanding systems using recurrent neural network. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 136–140.