Continuous Model Improvement for Language Understanding with Machine Translation

Abdalghani Abujabal
Amazon Alexa AI, Germany
abujabaa@amazon.de

Claudio Delli Bovi
Amazon Alexa AI, Germany
boviclau@amazon.de

Sungho Ryu
Amazon Alexa AI, Germany
shryu@amazon.de

Turan Gojayev
Amazon Alexa AI, Germany
tgojayev@amazon.de

Yannick Versley
Amazon Alexa AI, Germany
yversley@amazon.de

Fabian Triefenbach
Amazon Alexa AI, Germany
triefen@amazon.de

Abstract

Scaling conversational personal assistants to a multitude of languages puts high demands on collecting and labelling data, a setting in which cross-lingual learning techniques can help to reconcile the need for well-performing natural language understanding (NLU) with a desideratum to support many languages without incurring unacceptable cost. In this paper, we show that automatically annotating unlabeled utterances using machine translation in an offline fashion and adding them to the training data can improve performance for existing NLU features for low-resource languages, where a straightforward *translate-test* approach as considered in existing literature would fail the latency requirements of a live environment. We demonstrate the effectiveness of our method with intrinsic and extrinsic evaluation using a real-world commercial dialog system in German. We show that 56% of the resulting automatically labeled utterances had a perfect match with ground-truth labels. Moreover, we see significant performance improvements in an extrinsic evaluation settings when manually labeled data is available in small quantities.

1 Introduction

1.1 Motivation and Background

Voice-controlled personal assistants such as Amazon Alexa or Google Assistant have scaled to a large number of languages and see a constant influx of new functionalities that are exposed via the natural language interface. As a result, they have seen much interest around the development of multi-lingual and cross-lingual learning techniques that take this setting into consideration.

Beyond a setting where no target language data is available (*language expansion*, or *cross-lingual bootstrapping*), ongoing development also involves use cases where new functionalities from a resource-rich language (typically English) as the source language have to be integrated into existing training sets in the target language (*feature expansion*), or even settings where target language training data of current functionalities exists in small quantities, but accuracy falls short of its aim and an influx of unlabeled data in the target language exists and could be used for continuous model improvement (*feature improvement*). In this work, we consider feature improvement use case for natural language understanding (NLU) in low-resource languages. We define the task of NLU as the combination of: (1) Intent Classification (IC), which classifies an utterance into a fixed set of intent labels (e.g. `ApplianceOff`), and (2) Slot Labeling, which classifies slot values into a predefined set of slot types (e.g. `SongName`) (Weld et al., 2021). For example, as shown in Figure 1, a valid NLU annotation for the English utterance *"turn off the light in the hallway"* would be: `ApplianceOff`: (*turn*, `ActionTrigger`), (*off*, `ActionTrigger`), (*light*, `Device`), (*hallway*, `Location`), where `ApplianceOff` is the intent label, and `ActionTrigger`, `Device` and `Location` are the slot types. We leverage machine translation to automatically annotate unlabeled utterances with intent and slot labels. Collecting and labeling data for NLU is an expensive and time-consuming process, hardly scalable to an increasing number of languages without automation.

Utterance:	Turn	off	the	light	in	the	hallway
Intent:	ApplianceOff						
Slots:	ActionTrigger	O	Device	O	O	Location	

Figure 1: An example NLU annotated utterance. Non-slots are labeled with O (Other).

1.2 State of the Art and its Limitations

Many works on academic datasets naturally address the language expansion setting, including zero-shot

learning results, or involve a multilingual learning approach where similar amounts of training data for each of the languages are available from the start. However, we want to argue that the setting where target-language data is available but considerably smaller is particularly relevant in practice. Such an imbalance is often due cost considerations (manual annotation is expensive).

A first line of work on cross-lingual bootstrapping has combined annotation transfer with (to varying extent) either machine translation (MT) or parallel corpora. Generally, MT has been harnessed either in *translate-train* or *translate-test* settings. While in *translate-train*, source training data e.g., in English is translated into the target language (Gaspers et al., 2018), in *translate-test*, incoming unlabeled utterances in the target language are translated into the source language and then source NLU model is used to collect labels. For the feature improvement use case, on one hand, *translate-train* ignores the influx of unlabeled utterances in the target language. On the other hand, a *translate-test* approach is not directly applicable to production use in a conversational agent because a system with MT in the loop would fail the latency requirements for live use. As a consequence, we propose to use the label projection from the source language as a way to get more reliable labels than the existing target language model on less confident-cases, and augmenting the target language training data with these automatically labeled examples.

In sentiment classification, Mihalcea et al. (2007) compare translation of a lexicon with translating the training data (*translate-train*) or translating the data to be annotated (*translate-test*) for cross-lingual bootstrapping of sentiment classification. Akbik et al. (2015) investigate cross-lingual bootstrapping in the context of Semantic Role Labeling, where a parallel corpus is first annotated with English labels which are then projected and filtered to gain a target language training corpus. In dialogue systems and conversational agent training, He et al. (2013) show that adding some MT distortion to the source-language training data in a translate-test setting can be beneficial. Gaspers et al. (2018) show that a translate-train approach that uses machine translation in conjunction with filtering based on MT confidence can be successful in achieving a smaller error rate, with a combination of translated and target-language manually annotated data achieving the best possible error rate.

A second line of work concerns the use of shared representations across languages to cross-lingual transfer learning or learning of multilingual representations, as demonstrated by Upadhyay et al. (2018) who compare translate-train and translate-test approaches with zero-shot and minimally supervised multilingual approaches. It shows that the helpful bias from shared representations gives a boost in the minimally supervised setting but is especially helpful when very few target-language examples are available. Johnson et al. (2019) and Do et al. (2019) show that these effects generally also hold at a larger scale, and that training data selection also helps when transfer learning is used instead of machine translation in a translate-train setting.

Finally, and partially relevant for feature improvement when a smaller-than-source amount of target data is available, we have approaches that perform data augmentation on the smaller target-language training data: Malandrakis et al. (2019), and Jolly et al. (2020) explore the use of sentence-to-sentence paraphrasing and interpretation-to-sentence generation approaches to generate labeled paraphrases of conversational NLU training data.

1.3 Approach and Contribution

In this paper we investigate whether a *translate-test* approach of doing machine translation and annotation projection of target-language utterances with labels from the more resource-rich source language can be used in a *feature improvement* setting, where target-language training data is available but in smaller quantities than in the source language.

Our approach, depicted in Figure 2, makes use of MT in conjunction with an NLU model already trained for the source language to annotate unlabeled utterances. We assume that this reference NLU model was previously trained on the features of interest for the target language. Similar to Gaspers et al. (2018), we also assume access to an MT system trained on general-purpose parallel data, but instead of relying on MT from reference to target language, (*forward MT*), we consider MT in the opposite direction i.e. from target to reference language (*backward MT*). Our goal is to cheaply improve NLU features using readily available MT and NLU models. For example, we do not require in-domain MT model.

Experimentally, we considered a scenario with

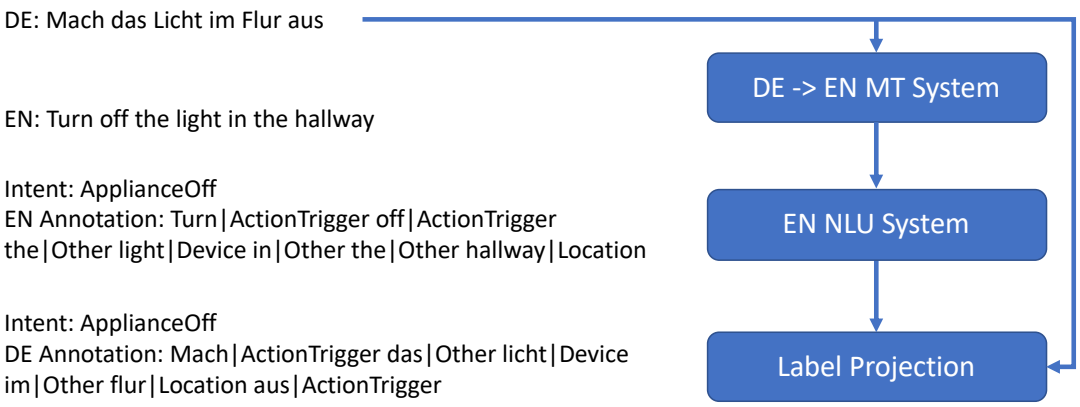

Figure 2: Given an unlabeled utterance in some target language e.g., German, our method translates it into a reference language e.g., English using MT, labels it with intent and slot types using an (EN) NLU model, and projects the labels back onto the unlabeled utterance.

English (EN) as reference language and German (DE) as target language, and carried out both an intrinsic and an extrinsic evaluation, where we selected a set of five diverse NLU features to improve. We compared against a baseline approach that generates synthetic training examples directly in the target language.

We demonstrate the effectiveness of our method using a real-world commercial dialog system in German. We show that 56% of the resulting automatically labeled utterances had a perfect match with ground-truth labels. We also show that using our method leads to 90% reduction in manually labeled data, while achieving better performance. In the remainder of the paper, Section 2 contains details on the methods used, whereas Section 3 describes our experimental setup. Section 4 discusses the results of our experiments.

2 Method

Given unlabeled utterances in a target language e.g. German (DE), for example *"mach das licht im flur aus"*, our goal is to automatically annotate them with an intent label, and slot types for every token, as shown in the example in Figure 1. To this end, we consider the pipeline shown in Figure 2, which consists of three components: (1) Machine translation system, (2) NLU model, and (3) Label projection model. First, the MT system translates the unlabeled utterances into a reference language e.g. English. For the German utterance above, a valid English translation would be *"turn off the light in the hallway"*. Note that we do not make any assumption on the architecture of the MT system, be it statistical or neural, or on the way it is trained. We assume, however, a label projection model trained on the same data as the MT system. In a standard MT bootstrapping setting (Gaspers et al., 2018) this is usually a word alignment model, either embedded in the MT system itself (as in phrase-based MT we used in Section 3) or trained as a stand-alone component. After translation, we use an English NLU model on the translated utterances in order to get predictions for the intent label and the slot types.[1] For the example above, the result of this step would be the following annotated utterance: [`ApplianceOff` *turn*/`ActionTrigger` *off*/`ActionTrigger` *the*/`Other` *light*/`Device` *in*/`Other` *the*/`Other` *hallway*/`Location`]. Finally, we use the word alignment model to project the slot types from the (EN) labeled utterances onto the unlabeled (DE) utterances. For example, if the two words *'light'* and *'licht'* are aligned, the slot label of *'light'* is copied over onto *'licht'*. In our experiments (Section 3), we make use of alignment models trained for the MT system to avoid building standalone alignment models. For the intent label, we simply copy it over from the English labeled utterance to the German unlabeled one.

For reasons of simplicity and better interpretability, we used statistical machine translation (SMT) as well as linear models (CRF and maximum entropy) for the NLU component, however we believe that the improvements gained with this method would carry over to a case where neural MT and transformer-based NLU components are used.

[1]Note that this NLU model is pre-trained independently and it is completely decoupled from our pipeline.

3 Experimental Setup

In our experiments, we translate target-language (German) utterances from live conversational agent usage using an existing MT system (§3.1), tag these using the English NLU system (§3.2) and project the labels back onto the target language using word alignments. We report results using first intrinsic evaluation (How well does the *translate-test* approach perform in labeling the utterances?) and then a full evaluation in a *feature improvement* setting, and we evaluate these using a Semantic Error Rate metric (SemER, §3.3).

3.1 MT System

We used an internal phrase-based MT system trained with Moses (Koehn et al., 2007). The system comprises a general-purpose MT model trained on DE-EN parallel data. We plan to investigate the usage of neural machine translation (NMT) models in the future. To better match the spoken user utterances of an NLU system, training data of the MT system is converted into *spoken form* using an internal written-to-spoken converter. For example, *"1994"* is converted to *"nineteen ninety four"*. The MT model was fine-tuned on 4K in-domain parallel utterances. To project slot type labels from the machine-translated English utterance (labeled by the English NLU model) to the unlabeled German utterance, we make use of the word alignment model trained for the MT system (Dyer et al., 2013). We opted for using a general-purpose MT model since it is readily available, and hence cheaper (as opposed to building in-domain MT model). Also, using phrase-based MT enabled us to leverage the word alignment model trained for MT for our label projection step.

3.2 NLU System

We used Conditional Random Fields (Lafferty et al., 2001) for slot labeling, and a Maximum Entropy classifier for the IC task (Berger et al., 1996). The English NLU system was trained on a large dataset of NLU-annotated utterances. The training data covers multiple domains e.g., `HomeAutomation`, with a diverse set of intents and slot types, with more than 200 intents and several hundreds of slot types. For example, intents like `PlayMusic` and slot types like `City` and `SongName`. The quality of the reference NLU model (e.g., English) is important for our pipeline to work. Our assumption is that English NLU models perform well, while NLU models for other languages still suffer (most industrial NLP applications support English pretty well).

Feature	#Auto labeled utterances	#Test utterances
DailyBriefing	21,894	3,530
PlayMusic	194,180	66,959
SendMessage	1,690	1,783
SmartHome	108,210	29,056
SetNotification	26,074	9,616

Table 1: The size of automatically labeled and test data for each feature.

3.3 SemER Evaluation Metric

Following Gaspers et al. (2018) we report the Semantic Error Rate (SemER), which is computed as follows:

$$SemER = \frac{\#(slots + intent\ errors)}{\#slots\ in\ reference + 1}$$

Errors correspond to the number of insertions, deletions and substitutions for slots and the intent in a recognized utterance.

Note that as the task of NLU is our main focus, we report evaluation metrics on the NLU rather intrinsically evaluating each component of our approach e.g., the MT model. We plan to invest in this direction in the future. Moreover, while intrinsic evaluation measures of individual components would assess their quality e.g., BLEU for MT, there is no correlation between these measures and NLU metrics. In other words, having higher BLEU scores does not necessarily mean lower SemER.

3.4 Utterance Dataset

To simulate a continuous model improvement scenario for DE, we selected a diverse set of features that belong to different domains:

1. *DailyBriefing*, which enables users to play daily briefing e.g., news,
2. *PlayMusic*, which enables playing music,
3. *SendMessage*, which allows users to exchange messages,
4. *SmartHome*, which enables users to control home appliances,
5. *SetNotification*, which enables users to set notifications and reminders.

Model	Size of training data	SemER (%)
DE	$6.4M$	41.4
DE_0.5	$3.7M$	**37.4**
DE_0.7	$3.2M$	37.8
DE_grammars	$1.0M$	57.0

Table 2: The effect of filtering the data based on NLU confidence. Using 0.5 achieved best results.

Features span across multiple intents with different slot labels. For example, *SmartHome* supports the intents of turning an appliance on and off, and supports the slot lables of appliance names and their locations. We assume that the five features have been just launched either using grammars, very little labeled data or using the approach of Gaspers et al. (2018). Our goal is to continuously improve performance on the five features using our method.

For each feature, we randomly selected 10,000 manually labeled utterances from its training data. Next, we generated five splits out of the 10,000 utterances: 100, 500, 1000, 5000 and 10,000. Each split corresponds to the size of data, for example, the split of 100 indicates that 100 manually labeled utterances are used. For each split, we trained two DE NLU models:

- **Baseline model**, which contains only manually labeled feature data, and
- **Combined model**, which contains both manually and automatically labeled feature data.

Note that the training data of the NLU models contain data for other features that were launched already. We report absolute SemER difference between the two models.

We collected 3,651,039 unlabeled DE utterances in order to run the MT-based automatic annotation. Table 1 shows the size of the automatically labeled data for each feature. We also collected test data for each feature (Table 1).

4 Results

4.1 Accuracy of Automatic Labeling

To intrinsically measure the accuracy of our method, we collected 1.2 million labeled utterances from features already launched in a real-world commercial dialog system in German, and simulated a scenario where the corresponding labels were not available. We then used our method to label them: we translated them into English, ran the English NLU model on them, and projected back all the predicted labels. We observed that 56.35% of the resulting automatically labeled utterances had a perfect match with ground-truth labels (i.e., they agreed on both the intent label and all the slot types), while 81.87% of them agreed on the intent only, with at least one unmatched slot type.

4.2 Effect of English NLU Confidence

We studied the effect of the English NLU model's prediction confidence. We collected 6.4M unlabeled German utterances and then used our method to annotate them. Each prediction (intent and slot labels) is associated with a score $\in [0, 1]$ that reflects the confidence of the English NLU model about the prediction. We then trained three DE NLU models: (1) DE, where confidence equals 0.0 i.e., 6.4M utterances are kept, (2) $DE\_0.5$, and (3) $DE\_0.7$, where utterances whose confidence score is greater than 0.5 and 0.7 are kept, respectively. The three models were tested on the same test set with 120K German utterances that were manually transcribed and annotated with intents and slot types. The test set spans multiple domains with different intents and slot types. As shown in Table 2, $DE\_0.5$ outperformed other baselines, indicating the importance of using NLU confidence scores. We attribute this to the fact that some translations are malformed, and hence incorrectly labeled by the English NLU model. When incorrect labels are propagated to the DE NLU model, they negatively impact performance. We set the EN NLU model's confidence score to 0.5 for the subsequent experiments.

We also trained an NLU model using randomly sampled utterances from manually curated grammars ($DE\_grammars$), which achieved 57.0 SemER and was outperformed by $DE\_0.5$, with 19.6 absolute SemER difference.

4.3 Feature Improvement

Table 3 shows the results on the five features, showing the SemER difference between a baseline (trained with the given number of hand-annotated utterances) and a version with our proposed method, combining the hand-annotated utterances with additional data which has been automatically labeled.

Combining manually and automatically labeled data improves performance across features and splits. The greatest gains are achieved for smaller splits i.e., 100 and 500, which suggest that our

Split	DailyBriefing	PlayMusic	SendMessage	SmartHome	SetNotification
100	**−38.65**	**−26.98**	**−13.25**	**−74.34**	**−19.42**
500	−10.72	−20.17	−1.47	−19.22	−9.97
1000	−7.24	−14.96	−0.88	−8.11	−7.5
5000	−1.69	−0.97	−0.21	+3.71	−0.18
10,000	−0.63	+2.72	−0.37	+3.12	−0.06

Table 3: SemER difference between the baseline and the combined model on the five features (lower is better). Across features, using automatically labeled data improved performance.

method is especially effective for an early feature improvement. For example, the difference in SemER between the baseline and the combined model is -38.65 on DailyBriefing at 100 split. For PlayMusic, SmartHome and SetNotification, the SemER value of the Combined model at 100 split is better than the one achieved by the baseline at 1000 split i.e., a reduction in labeled data of 90%.

As the size of manually labeled data increases (i.e., larger splits), the positive effect of the automatically labeled data decreases. For example, on DailyBriefing, the SemER difference between the baseline and Combined models is -0.63 absolute at 10,000 split. For the largest split at 10,000, the automatically labeled data hurts the performance for PlayMusic and SmartHome, with SemER difference of $+2.72$ and $+3.12$, respectively. This is largely due to cumulated errors in both the MT system and the label projection module, which inject noise in the downstream NLU task. To mitigate this, we are currently investigating ways to automatically combine training data with varying quality for NLU. We also carried out similar experiments to improve the same features in French and so far observed the same trends. We are planning to expand our evaluation to other languages.

5 Conclusion

This paper presents a new method to automatically annotate utterances with intents and slot types, leading to faster and cheaper early improvement of features. Our method harnesses existing MT, English NLU and word alignment models which have been trained on general-domain data but adapted to our specific use case through preprocessing and fine-tuning. Intrinsic evaluation results show that a *translate-test* approach is a viable way to get data labels in a way that is independent from the target language production system, whereas our extrinsic evaluation results suggest that the approach is especially useful when a given feature has not seen extensive use yet.

We plan to address in future work whether certain properties of a given feature can predict the viability of a *translate-test* approach in general and data augmentation with translated examples in particular, and whether the use of neural machine translation models would suggest modifications to this approach, as translations are often better but alignment results can be less clear-cut.

References

Alan Akbik, Laura Chiticariu, Marina Danilevsky, Yunyao Li, Shivakumar Vaithyanathan, and Huaiyu Zhu. 2015. Generating high quality proposition Banks for multilingual semantic role labeling. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 397–407, Beijing, China. Association for Computational Linguistics.

Adam L. Berger, Stephen Della Pietra, and Vincent J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Quynh Ngoc Thi Do and Judith Gaspers. 2019. Cross-lingual transfer learning with data selection for large-scale spoken language understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1455–1460. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 644–648. The Association for Computational Linguistics.

Judith Gaspers, Penny Karanasou, and Rajen Chatterjee. 2018. Selecting machine-translated data for quick bootstrapping of a natural language understanding system. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 3 (Industry Papers)*, pages 137–144. Association for Computational Linguistics.

X. He, L. Deng, D. Hakkani-Tur, and G. Tur. 2013. Multi-style adaptive training for robust cross-lingual spoken language understanding. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8342–8346.

Andrew Johnson, Penny Karanasou, Judith Gaspers, and Dietrich Klakow. 2019. Cross-lingual transfer learning for japanese named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 2 (Industry Papers)*, pages 182–189. Association for Computational Linguistics.

Shailza Jolly, Tobias Falke, Caglar Tirkaz, and Daniil Sorokin. 2020. Data-efficient paraphrase generation to bootstrap intent classification and slot labeling for new features in task-oriented dialog systems. In *Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*. The Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann.

Nikolaos Malandrakis, Minmin Shen, Anuj Kumar Goyal, Shuyang Gao, Abhishek Sethi, and Angeliki Metallinou. 2019. Controlled text generation for data augmentation in intelligent artificial agents. In *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019, Hong Kong, November 4, 2019*, pages 90–98. Association for Computational Linguistics.

Rada Mihalcea, Carmen Banea, and Janyce Wiebe. 2007. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 976–983, Prague, Czech Republic. Association for Computational Linguistics.

Shyam Upadhyay, Manaal Faruqui, Gökhan Tür, Dilek Hakkani-Tür, and Larry P. Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*, pages 6034–6038. IEEE.

Henry Weld, Xiaoqi Huang, Siqi Long, Josiah Poon, and Soyeon Caren Han. 2021. A survey of joint intent detection and slot-filling models in natural language understanding. *CoRR*, abs/2101.08091.