Efficient Intent Detection with Dual Sentence Encoders

github.com/PolyAI-LDN/polyai-models

Iñigo Casanueva,* **Tadas Temčinas**,* **Daniela Gerz, Matthew Henderson,** and **Ivan Vulić**
PolyAI Limited
London, United Kingdom
{inigo,dan,matt,ivan}@poly-ai.com

Abstract

Building conversational systems in new domains and with added functionality requires resource-efficient models that work under low-data regimes (i.e., in few-shot setups). Motivated by these requirements, we introduce intent detection methods backed by pretrained dual sentence encoders such as USE and ConveRT. We demonstrate the usefulness and wide applicability of the proposed intent detectors, showing that: **1)** they outperform intent detectors based on fine-tuning the full BERT-Large model or using BERT as a fixed black-box encoder on three diverse intent detection data sets; **2)** the gains are especially pronounced in few-shot setups (i.e., with only 10 or 30 annotated examples per intent); **3)** our intent detectors can be trained in a matter of minutes on a single CPU; and **4)** they are stable across different hyperparameter settings. In hope of facilitating and democratizing research focused on intention detection, we release our code, as well as a new challenging single-domain intent detection dataset comprising 13,083 annotated examples over 77 intents.

1 Introduction

Task-oriented conversational systems allow users to interact with computer applications through conversation in order to solve a particular task with well-defined semantics, such as booking restaurants, hotels and flights (Hemphill et al., 1990; Williams, 2012; El Asri et al., 2017), providing tourist information (Budzianowski et al., 2018), or automating customer support (Xu et al., 2017).

Intent detection is a vital component of any task-oriented conversational system (Hemphill et al., 1990; Coucke et al., 2018). In order to understand the user's current goal, the system must leverage its intent detector to classify the user's utterance (provided in varied natural language) into one of several predefined classes, that is, *intents*.[1] Scaling intent detectors (as well as conversational systems in general) to support new target domains and tasks is a very challenging and resource-intensive process (Wen et al., 2017; Rastogi et al., 2019). The need for expert domain knowledge and domain-specific labeled data still impedes quick and wide deployment of intent detectors. In other words, one crucial challenge is enabling effective intent detection in *low-data scenarios* typically met in commercial systems, with only several examples available per intent (i.e., the so-called *few-shot learning setups*).

Transfer learning on top of pretrained sentence encoders (Devlin et al., 2019; Liu et al., 2019b, *inter alia*) has now established as the mainstay paradigm aiming to mitigate the bottleneck with scarce in-domain data. However, directly applying the omnipresent sentence encoders such as BERT to intent detection may be sub-optimal. **1)** As shown by Henderson et al. (2019b), pretraining on a general language-modeling (LM) objective for conversational tasks is less effective than *conversational pretraining* based on the response selection task and conversational data (Henderson et al., 2019c; Mehri et al., 2019). **2)** Fine-tuning BERT and its variants is very resource-intensive as it assumes the adaptation of the full large model. Moreover, in few-shot setups fine-tuning may result in overfitting. From a commercial perspective, these properties lead to extremely slow, cumbersome, and expensive development cycles.

Therefore, in this work we propose to use efficient *dual sentence encoders* such as Universal Sentence Encoder (USE) (Cer et al., 2018) and ConveRT (Henderson et al., 2019b) to support intent detection. These models are in fact neural

*Equal contribution. TT is now at the Oxford University.

[1]For instance, in the e-banking domain intents can be *lost card* or *failed top-up* (see Table 2). The importance of intent detection is also illustrated by the fact that getting the intent wrong is the first point of failure of any conversational agent.

Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI, pages 38–45
July 9, 2020.

architectures tailored for modeling sentence pairs (Henderson et al., 2019c; Humeau et al., 2020), and are trained on a conversational response selection task. As such, they inherently encapsulate conversational knowledge needed for (few-shot) intent detection. We discuss their advantage over LM-based encoders, and empirically validate the usefulness of conversational pretraining for intent detection. We show that intent detectors based on fixed USE and ConveRT encodings outperform BERT-backed intent detectors across the board on three diverse intent detection datasets, with prominent gains especially in few-shot scenarios. Another advantage of dual models is their compactness:[2] we demonstrate that our state-of-the-art USE+ConveRT intent detectors can be trained even on a regular laptop's CPU in only several minutes.

We also show that intent classifiers based on dual sentence encoders are largely invariant to hyperparameter changes. This finding is extremely important for real-life low-data regimes: due to the invariance, the expensive hyperparameter tuning step can be bypassed, and a limited number of annotated examples can be used directly as additional training data (instead of held-out validation data).

Another contribution of this work is a new and challenging intent detection dataset in the banking domain, dubbed BANKING77. It follows the very recent endeavor of procuring high-quality intent detection data (Liu et al., 2019a; Larson et al., 2019), but is very different in nature than the other datasets. Unlike prior work which scatters a set of coarse-grained intents across a multitude of domains (i.e., 10+ domains, see Table 1 later), we present a challenging single-domain dataset comprising 13,083 examples over 77 fine-grained intents. We release the code as part of the growing PolyAI's repository: `github.com/PolyAI-LDN/polyai-models`. The BANKING77 dataset is available at: `github.com/PolyAI-LDN/task-specific-datasets`.

2 Methodology: Intent Detection with Dual Sentence Encoders

Pretrained Sentence Encoders. Large-scale pretrained models have benefited a wide spectrum of NLP applications immensely (Devlin et al., 2019; Liu et al., 2019b; Radford et al., 2019). Their core strength lies in the fact that, through consuming large general-purpose corpora during pretraining, they require smaller amounts of domain-specific training data to adapt to a particular task and/or domain (Ruder et al., 2019). The adaptation is typically achieved by adding a task-specific output layer to a large pretrained sentence encoder, and then fine-tuning the entire model (Devlin et al., 2019). However, the fine-tuning process is computationally intensive (Zafrir et al., 2019; Henderson et al., 2019b), and still requires sufficient task-specific data (Arase and Tsujii, 2019; Sanh et al., 2019). As such, the standard full fine-tuning approach is both unsustainable in terms of resource consumption (Strubell et al., 2019), as well as suboptimal for few-shot scenarios.

Dual Sentence Encoders and Conversational Pretraining. A recent branch of sentence encoders moves beyond the standard LM-based pretraining objective, and proposes an alternative objective: *conversational response selection*, typically on Reddit data (Al-Rfou et al., 2016; Henderson et al., 2019a). As empirically validated by Henderson et al. (2019c); Mehri et al. (2019), conversational (instead of LM-based) pretraining aligns better with conversational tasks such as dialog act prediction or next utterance generation.

Pretraining on response selection also allows for the use of efficient *dual* models: the neural response selection architectures are instantiated as dual-encoder networks that learn the interaction between inputs/contexts and their relevant (follow-up) responses. Through such response selection pretraining regimes they organically encode useful conversational cues in their representations.

In this work, we propose to use such efficient conversational dual models as the main source of (general-purpose) conversational knowledge to inform domain-specific intent detectors. We empirically demonstrate their benefits over other standard sentence encoders such as BERT in terms of **1)** performance, **2)** efficiency, and **3)** applicability in few-shot scenarios. We focus on two prominent dual models trained on the response selection task: Universal Sentence Encoder (USE) (Cer et al., 2018), and Conversational Representations from Transformers (ConveRT) (Henderson et al., 2019b). For further technical details regarding the two models, we refer the interested reader to the original work.

Intent Detection with dual Encoders. We implement a simple yet effective model (see §5 later) for intent detection which is based on the two dual models. Unlike with BERT, we do not fine-tune

[2] For instance, ConveRT is only 59MB in size, pretrained in less than a day on 12 GPUs (Henderson et al., 2019b).

the entire model, but use fixed sentence representations encoded by USE and ConveRT. We simply stack a Multi-Layer Perceptron (MLP) with a single hidden layer with ReLU non-linear activations (Maas et al., 2013) on top of the fixed representations, followed by a softmax layer for multi-class classification. This simple formulation also allows us to experiment with the combination of USE and ConveRT representations: we can feed the concatenated vectors to the same classification architecture without any further adjustment.

3 New Dataset: BANKING77

In spite of the crucial role of intent detection in any task-oriented conversational system, publicly available intent detection datasets are still few and far between, even for English. The previous standard datasets such as Web Apps, Ask Ubuntu, the Chatbot Corpus (Braun et al., 2017) or SNIPS (Coucke et al., 2018) are limited to only a small number of classes (< 10), which oversimplifies the intent detection task and does not emulate the true environment of commercial systems. Therefore, more recent work has recognized the need for improved and more challenging intent detection datasets. **1)** The dataset of Liu et al. (2019a), dubbed HWU64, contains 25,716 examples for 64 intents in 21 domains. **2)** The dataset of Larson et al. (2019), dubbed CLINC150, spans 150 intents and 23,700 examples across 10 domains.

However, the two recent English datasets are *multi-domain*, and the examples per each domain may not sufficiently capture the full complexity of each domain as encountered "in the wild". Therefore, to complement the recent effort on data collection for intent detection, we propose a new *single-domain* dataset: it provides a very fine-grained set of intents in a banking domain, not present in HWU64 and CLINC150. The new BANKING77 dataset comprises 13,083 customer service queries labeled with 77 intents. Its focus on fine-grained single-domain intent detection makes it complementary to the two other datasets: we believe that any comprehensive intent detection evaluation should involve both coarser-grained multi-domain datasets such as HWU64 and CLINC150, and a fine-grained single-domain dataset such as BANKING77. The data statistics are summarized in Table 1.

The single-domain focus of BANKING77 with a large number of intents makes it more challenging. Some intent categories partially overlap with others, which requires fine-grained decisions, see Table 2 (e.g., *reverted top-up* vs. *failed top-up*). Furthermore, as other examples from Table 2 suggest, it is not always possible to rely on the semantics of individual words to capture the correct intent.[3]

Dataset	Intents	Examples	Domains
HWU64	64	25,716	21
CLINC150	150	23,700	10
BANKING77 (ours)	77	13,083	1

Table 1: Intent detection datasets: key statistics.

4 Experimental Setup

Few-Shot Setups. We conduct all experiments on the three intent detection datasets described in §3. We are interested in wide-scale few-shot intent classification in particular: we argue that this setup most closely resembles the development process of a commercial conversational system, which typically starts with only a small number of data points when expanding to a new domain or task. We simulate such low-data settings by sampling smaller subsets from the full data. We experiment with setups where only 10 or 30 examples are available for each intent, while we use the same standard test sets for each experimental run.[4]

MLP Design. Unless stated otherwise (e.g., in experiments where we explicitly vary hyperparameters), for the MLP classifier, we use a single 512-dimensional hidden layer. We train with stochastic gradient descent (SGD), with the learning rate of 0.7 and linear decay. We rely on very aggressive dropout (0.75) and train for 500 iterations to reach convergence. We show how this training regime can improve the model's generalization capability, and we also probe its (in)susceptibility to diverse hyperparameter setups later in §5. Low-data settings are balanced, which is especially easy to guarantee in few-shot scenarios.

Models in Comparison. We compare intent detectors supported by the following pretrained sentence encoders. First, in the BERT-FIXED model we use pretrained BERT in the same way as dual encoders, in the so-called *feature mode*: we treat BERT as a black-box fixed encoder and use it to compute encodings/features for training the classifier.[5] We use

[3] The examples in BANKING77 are also longer on average (12 words) than in HWU64 (7 words) or CLINC150 (8).

[4] For reproducibility, we release all training subsets.

[5] We have also experimented with ELMo embeddings (Pe-

Intent Class	Example Utterance
Card Lost	*Could you assist me in finding my lost card?*
Link to Existing Card	*I found my lost card. Am I still able to use it?*
Reverted Top-up	*Hey, I thought my topup was all done but now the money is gone again – what's up with that?*
Failed Top-up	*Tell me why my topup wouldn't go through?*

Table 2: Intent classes and example utterances from BANKING77.

	BANKING77			CLINC150			HWU64		
Model	**10**	**30**	**Full**	**10**	**30**	**Full**	**10**	**30**	**Full**
BERT-FIXED	67.55	80.07	87.19	80.16	87.99	91.79	72.61	79.78	85.77
BERT-TUNED	83.42	90.03	**93.66**	91.93	95.49	96.93	84.86	88.27	92.10
USE	84.23	89.74	92.81	90.85	93.98	95.06	83.75	89.03	91.25
CONVERT	83.32	89.37	93.01	92.62	95.78	**97.16**	82.65	87.88	91.24
USE+CONVERT	**85.19**	**90.57**	93.36	**93.26**	**96.13**	**97.16**	**85.83**	**90.16**	**92.62**

Table 3: Accuracy scores ($\times 100\%$) on all three intent detection data sets with varying number of training examples (**10** examples per intent; **30** examples per intent; **Full** training data). The peak scores per column are in bold.

the mean-pooled "sequence ouput" (i.e., the pooled mean of the sub-word embeddings) as the sentence representation.[6] In the BERT-TUNED model, we rely on the standard BERT-based fine-tuning regime for classification tasks (Devlin et al., 2019) which adapts the full model. We train a softmax layer on top of the [CLS] token output. We use the Adam optimizer with weight decay and a learning rate of 4×10^{-4}. For low-data (10 examples per intent), mid-data (30 examples) and full-data settings we train for 50, 18, and 5 epochs, respectively, which is sufficient for the model to converge, while avoiding overfitting or catastrophic forgetting.

We use the two publicly available pretrained dual encoders: **1)** the multilingual large variant of USE (Yang et al., 2019),[7] and **2)** the single-context CONVERT model trained on the full 2015-2019 Reddit data comprising 654M *(context, response)* training pairs (Henderson et al., 2019b).[8] In all experimental runs, we compare against the pretrained cased BERT-large model: 24 Transformer layers, embedding dimensionality 1024, and a total of 340M parameters. Note that e.g. ConveRT is much lighter in its design and is also pretrained more quickly than BERT (Henderson et al., 2019b): it relies on 6 Transfomer layers with embedding dimensionality of 512. We report accuracy as the main evaluation measure for all experimental runs.

ters et al., 2018) in the same feature mode, but they are consistently outperformed by all other models in comparison.

[6]This performed slightly better than using the [CLS] token embedding as sentence representation.

[7]https://tfhub.dev/google/universal-sentence-encoder-multilingual-large/1

[8]https://github.com/PolyAI-LDN/polyai-models

5 Results and Discussion

Table 3 summarizes the main results; we show the accuracy scores of all models on all three datasets, and for different training data setups. As one crucial finding, we report competitive performance of intent detectors based on the two dual models, and their relative performance seems to also depend on the dataset at hand: USE has a slight edge over CONVERT on HWU64, but the opposite holds on CLINC150. The design based on fixed sentence representations, however, allows for the straightforward combination of USE and CONVERT. The results suggest that the two dual models in fact capture complementary information, as the combined USE+CONVERT-based intent detectors result in peak performance across the board. As discussed later, due to its pretraining objective, BERT is competitive only in its fine-tuning mode of usage, and cannot match other two sentence encoders in the feature-based (i.e., fixed) usage mode.

Few-Shot Scenarios. The focus of this work is on low-data few-shot scenarios often met in production, where only a handful of annotated examples per intent are available. The usefulness of dual sentence encoders comes to the fore especially in this setup: 1) the results indicate gains over the fine-tuned BERT model especially for few-shot scenarios, and the gains are more pronounced in our "fewest-shot" setup (with only 10 annotated examples per intent). The respective improvements of USE+CONVERT over BERT-TUNED are +1.77, +1.33, and +0.97 for BANKING77, CLINC150, and HWU64 (10 examples per intent), and we also see

	BANKING77		CLINC150		HWU64	
Model	**10**	**Full**	**10**	**Full**	**10**	**Full**
BERT-FIXED	64.9 (67.8) [57.0]	86.2 (88.4) [74.9]	78.1 (80.6) [70.2]	91.2 (92.6) [84.7]	71.5 (72.8) [68.0]	85.9 (86.8) [81.5]
USE	83.9 (84.4) [83.0]	92.6 (92.9) [91.4]	90.6 (91.0) [89.9]	95.0 (95.3) [93.9]	83.6 (83.9) [83.0]	91.6 (92.1) [90.7]
CONVERT	83.1 (83.4) [82.4]	92.6 (93.0) [91.6]	92.4 (92.8) [92.0]	97.1 (97.2) [96.3]	82.5 (83.1) [82.0]	91.3 (91.6) [90.8]
USE+CONVERT	85.2 (85.5) [84.8]	93.3 (93.5) [92.8]	93.2 (93.5) [92.8]	97.0 (97.2) [96.5]	85.9 (86.2) [85.7]	92.5 (92.8) [91.6]

Table 4: Variation in accuracy scores ($\times 100\%$) with different hyperparameter regimes for all the models in comparison and on all three datasets. **10** again means 10 training examples per intent as opposed to **Full** training data. The scores are provided as *avg (max) [min]*: *avg* is the average over all runs with different hyperparameter settings for each encoder model and each setup, *max* and *min* are the respective maximum and minimum scores.

Encoder	CPU	GPU
BERT (Large)	2.4	235.9
USE	53.5	785.4
CONVERT	58.3	866.7

Table 5: Average number of sentences encoded *per second* with the three sentence encoders. The data is fed to each encoder in batches of 15 sentences.

Classifer	CPU	GPU	TPU
BERT-TUNED	n/a	n/a	567s
USE	65s	57s	n/a
CONVERT	73s	53s	n/a

Table 6: Time to train and evaluate an intent classification model based on two dual models and fine-tuning BERT on BANKING77 in a few-shot scenario with 10 examples per intent. The CPU is a 2.3 GHz Dual-Core Intel Core i5. The GPU is a GeForce RTX 2080 Ti, 11 GB. The TPU is a v2-8, 8 cores, 64 GB.

better results with the combined model when 30 examples per intent are available on all three datasets. Overall, this proves the suitability of dual sentence encoders for the few-shot intent classification task.

Invariance to Hyperparameters. A prominent risk in few-shot setups concerns overfitting to small data sets (Srivastava et al., 2014; Olson et al., 2018). Another issue concerns the sheer lack of training data, which gets even more pronounced if a subset of the (already scarce) data must be reserved for validation and hyper-parameter tuning. Therefore, a desirable property of any few-shot intent detector is its invariance to hyperparameters and, consequently, its off-the-shelf usage without further tuning on the validation set. This effectively means that one could use all available annotated examples directly for training. In order to increase the reliability of the intent detectors and prevent overfitting in few-shot scenarios, we suggest to use the aggressive dropout regularization (i.e., the dropout rate is 0.75), and a very large number of iterations (500), see §4.

We now demonstrate that the intent detectors based on dual encoders are very robust with respect to different hyper-parameter choices, starting from this basic assumption that a high number of iterations and high dropout rates r are needed. For each classifier, we fix the *base/pivot* configuration from §4: the number of hidden layers is $H = 1$, its dimensionality is $h = 512$, the SGD optimizer is used with the learning rate of 0.75. Starting from the pivot configuration, we create other configurations by altering one hyper-parameter at the time from the pivot. We probe the following values: $r = \{0.75, 0.5, 0.25\}$, $H = \{0, 1, 2\}$, $h = \{128, 256, 512, 1024\}$, and we also try out all the configurations with another optimizer: Adam with the linearly decaying learning rate of 4×10^{-4}.

The results with all hyperparameter configs are summarized in Table 4. They suggest that intent detectors based on dual models are indeed very robust. Importantly, we do not observe any experimental run which results in substantially lower performance with these models. In general, the peak scores with dual-based models are reported with higher r rates (0.75), and with larger hidden layer sizes h (1,024). On the other side of the spectrum are variants with lower r rates (0.25) and smaller h-s (128). However, the fluctuation in scores is not large, as illustrated by the results in Table 4. This finding does not hold for BERT-FIXED where in Table 4 we do observe "outlier" runs with substantially lower performance compared to its peak and average scores. Finally, it is also important to note BERT-TUNED does not converge to a good solution for 2% of the runs with different seeds, and such runs are not included in the final reported numbers with that baseline in Table 3.

Resource Efficiency. Besides strong performance established in Table 3 and increased stability (see Table 4), another advantage of the two dual models is their *encoding efficiency*. In Table 5 we report

the average times needed by each fixed encoder to encode sentences fed in the batches of size 15 on both CPU (2.3 GHz Dual-Core Intel Core i5) and GPU (GeForce RTX 2080 Ti, 11 GB). The encoding times reveal that BERT, when used as a sentence encoder, is around 20 times slower on the CPU and roughly 3 times slower on the GPU.[9]

Furthermore, in Table 6 we present the time required to train and evaluate an intent classification model for BANKING77 in the lowest-data regime (10 instances per intent).[10] Note that the time reduction on GPU over CPU for the few-shot scenario is mostly due to the reduced encoding time on GPU (see Table 5 again). However, when operating in the *Full* data regime, the benefits of GPU training vanish: using a neural net with a single hidden layer the overhead of the GPU usage is higher than the speed-up achieved due to faster encoding and network computations. Crucially, the reported training and execution times clearly indicate that effective intent detectors based on pretrained dual models can be constructed even without large resource demands and can run even on CPUs, without huge models that require GPUs or TPUs. In sum, we hope that our findings related to improved resource efficiency of dual models, as well as the shared code will facilitate further and wider research focused on the intent classification task.

Further Discussion. The results from Tables 3 and 4 show that transferring representations from conversational pretraining based on the response selection task (and conversational data) is useful for conversational tasks such as intent detection. This corroborates the main findings from prior work (Humeau et al., 2020; Henderson et al., 2019b). The results also suggest that using the current pretrained BERT as an off-the-shelf sentence encoder is sub-optimal for an application such as intent detection: BERT is much more powerful when used in the fine-tuning mode instead of the less expensive "feature-based" mode (Peters et al., 2019). This might be due to its pretraining LM objective: while both USE and ConveRT are forced to reason at the level of full sentences during the response selection pretraining, BERT is primarily a (local) language model. It seems that the next sentence prediction objective is not sufficient to learn a universal sentence encoder which can be applied off-the-shelf to unseen sentences in conversational tasks (Mehri et al., 2019). However, BERT's competitive performance in the fine-tuning mode, at least in the *Full* data scenarios, suggests that it still captures knowledge which is useful for intent detection. Given strong performance of both fine-tuned BERT and dual models in the intent detection task, in future work we plan to investigate hybrid strategies that combine dual sentence encoders and LM-based encoders. Note that it is also possible to combine BERT-FIXED with the two dual encoders, but such ensembles, besides yielding reduced performance, also substantially increase training times (see again Table 5).

We also believe that further gains can be achieved by increasing the overall size and depth of dual models such as ConveRT, but this comes at the expense of its efficiency and training speed: note that the current architecture of ConveRT relies on only 6 Transformer layers and embedding dimensionality of 512 (cf., BERT-Large with 24 layers and 1024-dim embeddings).

6 Conclusion

We have presented intent classification models that rely on sentence encoders which were pretrained on a conversational response selection task. We have demonstrated that using dual encoder models such as USE and ConveRT yield state-of-the-art intent classification results on three diverse intent classification data sets in English. One of these data sets is another contribution of this work: we have proposed a fine-grained single-domain data set spanning 13,083 annotated examples across 77 intents in the banking domain.

The gains with the proposed models over fully fine-tuned BERT-based classifiers are especially pronounced in few-shot scenarios, typically encountered in commercial systems, where only a small set of annotated examples per intent can be guaranteed. Crucially, we have shown that the proposed intent classifiers are extremely lightweight in terms of resources, which makes them widely usable: they can be trained on a standard laptop's CPU in several minutes. This property holds promise to facilitate the development of intent classifiers even without access to large computational

[9] We provide a *colab* script to reproduce these experiments.

[10] Note that we cannot evaluate BERT-TUNED on GPU as it runs out of memory. Similar problems were reported in prior work; see https://github.com/google-research/bert/blob/master/README.md#squad-11 for a reference. USE and CONVERT cannot be evaluated on TPUs as they currently lack TPU-specific code.

resources, which in turn also increases equality and fairness in research (Strubell et al., 2019).

In future work we will port the efficient intent detectors based on dual encoders to other languages, leveraging multilingual pretrained representations (Chidambaram et al., 2019). This work has also empirically validated that there is still ample room for improvement in the intent detection task especially in low-data regimes. Therefore, similar to recent work (Upadhyay et al., 2018; Khalil et al., 2019; Liu et al., 2019c), we will also investigate how to transfer intent detectors to low-resource target languages in few-shot and zero-shot scenarios. We also plan to extend the models to handle out-of-scope prediction (Larson et al., 2019).

We have released the code and the data sets online at: `github.com/PolyAI-LDN/polyai-models`.

Acknowledgments

We thank our colleagues at PolyAI, especially Paweł Budzianowski, Sam Coope, and Shawn Wen for many fruitful discussions. We also thank the anonymous reviewers for their helpful suggestions.

References

Rami Al-Rfou, Marc Pickett, Javier Snaider, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2016. Conversational contextual cues: The case of personalization and history for response ranking. *CoRR*, abs/1606.00372.

Yuki Arase and Jun'ichi Tsujii. 2019. Transfer fine-tuning: A BERT case study. In *Proceedings of EMNLP-IJCNLP*, pages 5392–5403.

Daniel Braun, Adrian Hernandez Mendez, Florian Matthes, and Manfred Langen. 2017. Evaluating natural language understanding services for conversational question answering systems. In *Proceedings of SIGDIAL*, pages 174–185.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of EMNLP*, pages 5016–5026.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Muthuraman Chidambaram, Yinfei Yang, Daniel Cer, Steve Yuan, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Learning cross-lingual sentence representations via a multi-task dual-encoder model. In *Proceedings of the 4th Workshop on Representation Learning for NLP*, pages 250–259.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips Voice Platform: An embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, pages 12–16.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: A corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of SIGDIAL*, pages 207–219.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS Spoken Language Systems Pilot Corpus. In *Proceedings of the Workshop on Speech and Natural Language*, pages 96–101.

Matthew Henderson, Pawel Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulić, and Tsung-Hsien Wen. 2019a. A repository of conversational datasets. In *Proceedings of the 1st Workshop on Natural Language Processing for Conversational AI*, pages 1–10.

Matthew Henderson, Iñigo Casanueva, Nikola Mrkšić, Pei-Hao Su, Ivan Vulić, et al. 2019b. ConveRT: Efficient and accurate conversational representations from transformers. *arXiv preprint arXiv:1911.03688*.

Matthew Henderson, Ivan Vulić, Daniela Gerz, Iñigo Casanueva, Paweł Budzianowski, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić, and Pei-Hao Su. 2019c. Training neural response selection for task-oriented dialogue systems. In *Proceedings of ACL*, pages 5392–5404.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *Proceedings of ICLR*, volume abs/1905.01969.

Talaat Khalil, Kornel Kiełczewski, Georgios Christos Chouliaras, Amina Keldibek, and Maarten Versteegh. 2019. Cross-lingual intent classification in a low resource industrial setting. In *Proceedings of EMNLP-IJCNLP*, pages 6418–6423.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of EMNLP-IJCNLP*, pages 1311–1316.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019a. Benchmarking natural language understanding services for building conversational agents. In *Proceedings of IWSDS*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Zihan Liu, Jamin Shin, Yan Xu, Genta Indra Winata, Peng Xu, Andrea Madotto, and Pascale Fung. 2019c. Zero-shot cross-lingual dialogue systems with transferable latent variables. In *Proceedings of EMNLP-IJCNLP*, pages 1297–1303.

Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of ICML*.

Shikib Mehri, Evgeniia Razumovskaia, Tiancheng Zhao, and Maxine Eskenazi. 2019. Pretraining methods for dialog context representation learning. In *Proceedings of ACL*, pages 3836–3845.

Matthew Olson, Abraham J. Wyner, and Richard Berk. 2018. Modern neural networks generalize on small data sets. In *Proceedings of NeurIPS*, pages 3623–3632.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. 2019. To tune or not to tune? Adapting pretrained representations to diverse tasks. In *Proceedings of the 4th Workshop on Representation Learning for NLP*, pages 7–14.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2019. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. *arXiv preprint arXiv:1909.05855*.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of NAACL-HLT: Tutorials*, pages 15–18.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of ACL*, pages 3645–3650.

Shyam Upadhyay, Manaal Faruqui, Gökhan Tür, Dilek Hakkani-Tür, and Larry P. Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. In *Proceedings of ICASSP*, pages 6034–6038.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of EACL*, pages 438–449.

Jason Williams. 2012. A critical analysis of two statistical spoken dialog systems in public use. In *Proceedings of SLT*.

Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. 2017. A new chatbot for customer service on social media. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3506–3510.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernández Ábrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Multilingual universal sentence encoder for semantic retrieval. *CoRR*, abs/1907.04307.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Wasserblat. 2019. Q8BERT: Quantized 8bit BERT. *CoRR*, abs/1910.06188.