Cascading Adaptors to Leverage English Data to Improve Performance of Question Answering for Low-Resource Languages

Hariom A. Pandya [*] [†]
Dharmsinh Desai University
Nadiad-Gujarat(India)
pandya.hariom@gmail.com

Bhavik Ardeshna [*†]
Dharmsinh Desai University
Nadiad-Gujarat(India)
ardeshnabhavik@gmail.com

Brijesh S. Bhatt
Dharmsinh Desai University
Nadiad-Gujarat(India)
brij.ce@ddu.ac.in

Abstract

Transformer based architectures have shown notable results on many down streaming tasks including question answering. The availability of data, on the other hand, impedes obtaining legitimate performance for low-resource languages. In this paper, we investigate the applicability of pre-trained multilingual models to improve the performance of question answering in low-resource languages. We tested four combinations of language and task adapters using multilingual transformer architectures on seven languages similar to MLQA dataset. Additionally, we have also proposed zero-shot transfer learning of low-resource question answering using language and task adapters. We observed that stacking the language and the task adapters improves the multilingual transformer models' performance significantly for low-resource languages. Our code and trained models are available at : https://github.com/CALEDIPQALL/

1 Introduction

Last few years have seen emergence of transformer based pretrained models like BERT(Devlin et al., 2019), XLNet(Yang et al., 2019), T5(Raffel et al., 2020), XLM-RoBERTa(Conneau et al., 2020) etc. The pretrained models have shown significant improvement in various downstream tasks like question answering, NER, Machine translation and speech recognition(Delobelle et al., 2020; Pires et al., 2019; Pfeiffer et al., 2020a; Pires et al., 2019; Pandya and Bhatt, 2021; Saha et al., 2021; Murthy et al., 2019; Park et al., 2008; Raffel et al., 2020).

The emergence of multilingual models: mBERT (Devlin et al., 2019) and XLM-RoBERTa(Conneau et al., 2020) made it possible to leverage English data to improve the performance of low-resource languages. In this paper, we continue to investigate the effectiveness of multilingual pretrained transformer models in improving the performance of question answering systems in a low-resource setup using the cascading of language and task adapters(Pfeiffer et al., 2021, 2020a; Bapna and Firat, 2019). Our work contributes by evaluating cross-lingual performance in seven languages - Hindi, Arabic, German, Spanish, English, Vietnamese and Simplified Chinese. Our models are evaluated on the combination of XQuAD(Artetxe et al., 2020) and MLQA(Lewis et al., 2020) datasets which are similar to SQuAD (Rajpurkar et al., 2016) .

To this end, our contributions are as follows:

- We have trained multilingual variants of transformers, namely mBert and XLM-RoBERTa with a QA dataset in seven languages. Both the MLQA and XQuAD datasets contain validation and test sets for the above languages but not the training set. To finetune the model we have combined the test set of XQuAD and MLQA datasets and evaluated the model with the MLQA development dataset as the test dataset. By splitting the dataset in this way we can get train and test data with the considerable length for low-resource languages which helped us to conduct various experiments. Table 1 highlights the size of our train and test set for all the above-mentioned languages.
- We exhaustively analysed the fine-tuned models by evaluating them with the tasks adapter[1] (Pfeiffer et al., 2021, 2020a). We conducted the experiments in two different setups, Houlsby(Houlsby et al., 2019) and Pfeif-

[*]Equal contribution
[†]Corresponding Authors

[1]Pre-trained task adapters from https://adapterhub.ml/explore/qa/squad1/

Proceedings of the 18th International Conference on Natural Language Processing, pages 544–549
Silchar, India. December 16 - 19, 2021.

	Hindi	German	Spanish	Arabic	Chinese	Vietnamese	English
Train	6854	5707	6443	6525	6327	6685	12780
Test	507	512	500	517	504	511	1148

Table 1: Size of the train and test set used in the experiments. The MLQA(Lewis et al., 2020) test and XQuAD(Artetxe et al., 2020) datasets are used for fine-tuning the model and for testing purpose MLQA devset is used for all languages to maintain consistency.

fer(Pfeiffer et al., 2021, 2020b). These two setups enabled us to compare our language model variants with their multilingual counterparts and understand the different factors that lead to better results on the downstream tasks.

- We have also attempted a series of two different experiments by stacking language adapters and task adapter[2] in different ways. We first analyze the fine-tuned model by stacking language-specific adapter with the XLM-RoBERTa$_{base}$ [3]. After fine-tuning the language-specific adapter we augment the task-specific adapter upon the previously fine-tuned language adapter. We analyze both the experiments separately and conclude that multiple adapters with the transformer-based model perform notably better.

- Due to limited training, the transfer-learning performance of the transformer is poor on the low-resource languages as well as on the languages unseen during the pretraining(Kakwani et al., 2020). The multi-task adapter (MAD-X) (Pfeiffer et al., 2020b) outperforms the state-of-the-art models in cross-lingual transfer across a representative set of typologically diverse languages on question answering. To avoid the training of model individually for multiple languages while maintaining the performance, we used cross-lingual transfer by switching heads of language adapter from the source language to the target language.

2 Proposed Approach

In this section we describe our approach of training the task adapter and the language adapters in 4 different setup.

2.1 Cross-Lingual Tuning of Task Adapter and Language Adapters

Task-Specific Cross-Lingual Transfer: We have used two different configurations for fine-tuning the task-specific adapter for cross-lingual transfer in low-resource languages (Pfeiffer et al., 2021; Houlsby et al., 2019). We have fine-tuned XLM-RoBERTa$_{base}$ for multiple languages with the question answering corpora. We calculated the F1-Score, Exact Match, Jaccard [4] , and WER (Word Error Rate)(Park et al., 2008) [5] for the test dataset.

Adapting Cross-Lingual learning using Language-Specific Model: We used the language adapter trained using unlabelled data on MLM objective. It makes the pretrained multilingual model more suitable for the specific language with its improved language understanding. We perform the downstream task by stacking specific language adapter with the XLM-RoBERTa$_{base}$ and used recent efficient adapter architecture proposed by pfeiffer et al. (Pfeiffer et al., 2021).

After fine-tuning task-specific adapter and language-specific adapters individually with the different low-resource languages, we observed that by stacking task adapter and language adapters together with the transformer model the performance improved significantly. For each language available in MLQA, we fine-tuned a task adapter using a corresponding question answering dataset.

2.2 Multi-Task Adapter for Cross-Lingual Transfer

The adapter-based MAD-X framework (Pfeiffer et al., 2020b) enables learning language-specific and task-specific transformations in a modular and parameter-efficient way. Our method of using MAD-X is as follows:

[2]Pre-Trained Language Adapters from https://adapterhub.ml/explore/text_lang/

[3]XLM-RoBERTa$_{base}$ https://huggingface.co/deepset/roberta-base-squad2

[4]Jaccard score https://en.wikipedia.org/wiki/Jaccard_index

[5]WER score https://en.wikipedia.org/wiki/Word_error_rate

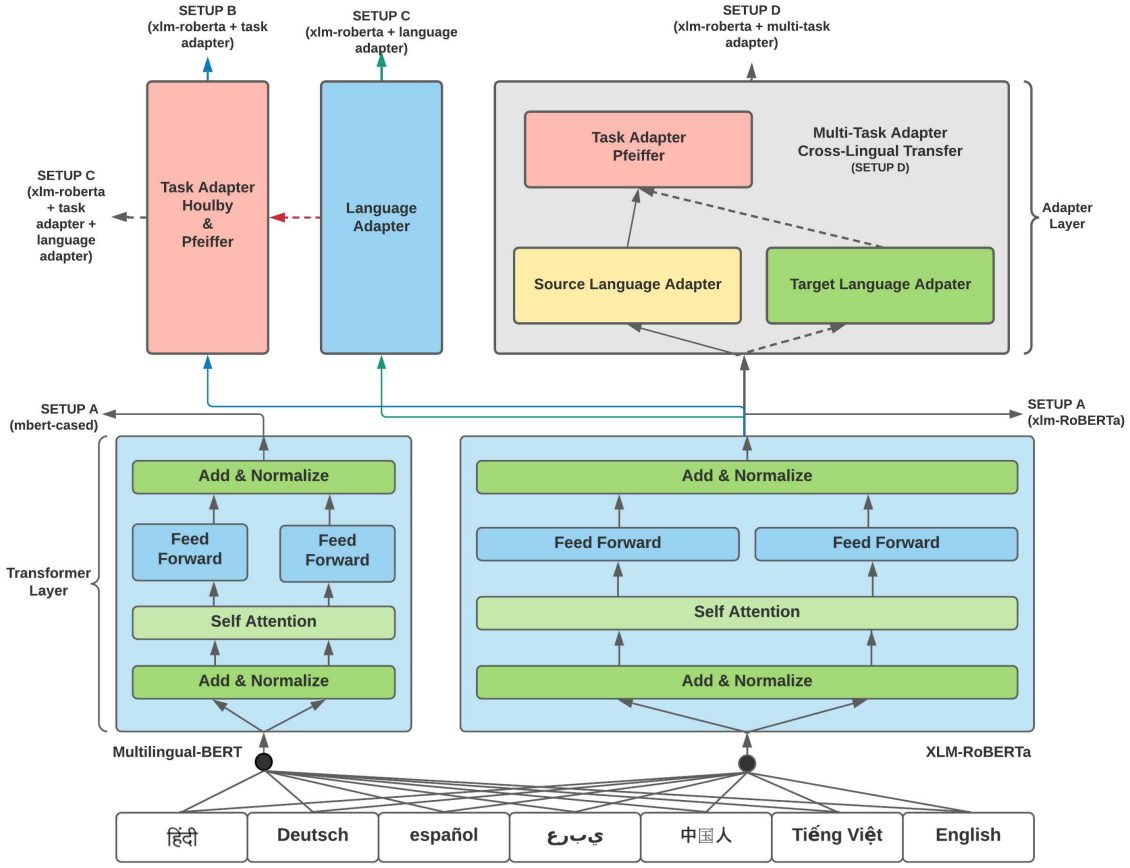

Figure 1: Experimental architecture- SETUP A: mBERT and XLM-R for QA, SETUP B: XLM-R with task adapters setup, SETUP C: XLM-R with language and language + task adapter and SETUP D: MAD-X setup for XLM-R

1. We have used pre-trained language adapters[6] for the source and target language on a language modeling task.

2. Train a task adapter on the target task dataset. This task adapter is stacked upon the previously trained language adapter. During this step, only the weights of the task adapter are updated.

3. Next, in zero-shot cross-lingual transfer step, we replaced the source language adapter with the target language adapter while keeping the stacked task adapter.

3 Experimental setups

We have performed 4 different analysis as represented in Figure 1. Details of all 4 setups are shown below:

[6]from https://adapterhub.ml/

3.1 Setup A

Here, we evaluated mBERT, XLM-Roberta$_{base}$ and XLM-Roberta$_{large}$ models on downstream tasks with the training dataset, which is specific to the individual language variant. The EM and F1 score for all languages are shown in Table 2.

Here, the interpretation of the matrix is F1/EM and it is same for rest of the Setups. For Example, in Table 2 first entry 56.25/39.45 indicates, for the Hindi test set, the F1score=56.25 and EM=39.45 is achieved using mBERT transformer model.

3.2 Setup B

After fine-tuning the transformer model, We have evaluated XLM-RoBERTa$_{base}$ with the task-specific adapter on downstream tasks under two training settings: Houlby(Houlsby et al., 2019) and Pfeiffer(Pfeiffer et al., 2021). While fine-tuning, the weights of only the task adapter get updated and the model weights are kept unchanged. This setup enables the scalable sharing of the task adapter model particularly in low-resource scenarios. Pre-

	Hindi	German	Spanish	Arabic	Chinese	Vietnamese	English
mBERT	56.25 / 39.45	52.99 / 38.09	59.89 / 40.4	51.28 / 31.33	41.86 / 41.07	59.52 / 39.73	77.86 / 63.85
XLM-RoBERTa$_{base}$	64.49 / 48.32	60.74 / 45.31	68.99 / 47.6	58.07 / 39.65	45.37 / 44.24	68.19 / 48.53	81.29 / 68.64
XLM-RoBERTa$_{large}$	73.37 / 56.02	70.57 / 53.32	76.32 / 54.2	67.15 / 47.78	49.94 / 49.21	73.78 / 54.21	85.98 / 74.39

Table 2: F1 score and Exact Match on the test set for the Setup A on multilingual-BERT and XLM-RoBERTa.

	Hindi	German	Spanish	Arabic	Chinese	Vietnamese	English
Task Adapter (*Houlby*)	64.12 / 47.73	**60.95 / 44.53**	68.48 / 46.6	**58.13 / 38.49**	**44.38 / 43.25**	68.39 / 48.34	80.86 / 68.29
Task Adapter (*Pfeiffer*)	**65.7 / 49.9**	60.53 / 44.14	**69.09 / 48**	55.97 / 37.14	44.05 / 43.05	**68.46 / 48.53**	**81.23 / 68.64**

Table 3: F1 score and Exact Match for the xlm-roberta with Task Adapter (Setup B). We bold the best results.

trained task-specific adapters: Houlby[7] and Pfeiffer[8] are taken with predefined conditions. The EM and F1 score for all languages are shown in Table 3.

3.3 Setup C

The language adapters are used to learn language-specific transformations (Pfeiffer et al., 2020b). After being trained on a language modeling task, a language adapter can be stacked before a task adapter for training on a downstream task. To perform zero-shot cross-lingual transfer, one language adapter can be replaced by another. In terms of architecture, language adapters are largely similar to task adapters, except for an additional invertible adapter layer after the embedding layer.

In this setup, we have evaluated each language-specific adapter[9] by stacking it on the XLM-RoBERTa model. In the second phase, we stacked the task-specific adapter and language-specific adapter on the XLM-RoBERTa model. The EM and F1 score for the language adapter and the task + language adapter fusion are shown in Table 4.

3.4 Setup D

Here, we have cascaded the multi-task adapters(Pfeiffer et al., 2020b) to leverage the high-resource dataset to improve the performance of the low-resource language. We stacked the fine-tuned task-specific adapter upon the language-specific adapter and XLM-RoBERTa (shown in figure 1). After fine-tuning with high resource language, we performed zero-shot cross-lingual transfer by switching the source language adapter with the target language adapters. Our results for multi-task adapters are highlighted in the Table 6.

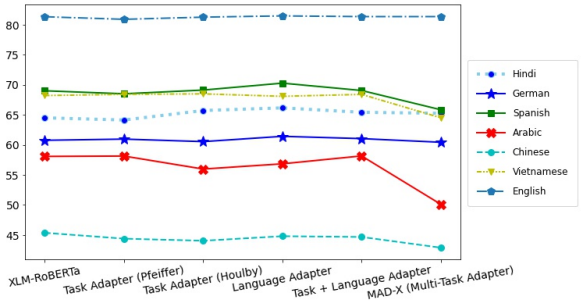

Figure 2: The performance of the different heads. The Y-axis here denotes the F1 score

Table 5 shows Jaccard and WER score for all four setups while the Figure 2 represents the F1 score of our models on all the languages.

4 Observations

To study the impact of the task adapter and the language adapters, we have conducted experiments as shown in Setup B and Setup C. Our observations from Table 3 and Table 4 indicates that the trained language adapter (Setup C: language adapters only) improves the performance for Hindi, German, Spanish, Chinese and English languages over the usage of task adapter(Setup B). However, instead of using language adapters only the stack of task and the language adapters lower EM and F1 score for languages other than Arabic.

We have compared two task adapter architectures and noted that the usage of different task adapter architectures have negligible performance impact on majority of the languages. As a result, no clear distinction can be drawn from this observation, which can be used to guide future research.

High-resource languages that use the Right-to-Left (RTL) scripting approach dominate the training of pretrained transformer models. The Arabic language follows Left-to-right(LTR) scripting

[7]Available at https://adapterhub.ml/adapters/ukp/roberta-base_qa_squad1_houlsby/

[8]https://adapterhub.ml/adapters/ukp/roberta-base_qa_squad1_pfeiffer/

[9]Available at https://adapterhub.ml/explore/text_lang/

	Hindi	German	Spanish	Arabic	Chinese	Vietnamese	English
Language Adapter	**66.14 / 49.11**	**61.41 / 45.9**	**70.25 / 49.6**	56.84 / 37.52	**44.82 / 43.85**	68.06 / **49.31**	**81.43** / 68.64
Task + Language Adapter	65.39 / 48.72	61.03 / 45.51	69.03 / 47.6	**58.15 / 38.29**	44.68 / 43.45	**68.39** / 48.14	81.31 / **68.9**

Table 4: F1 score and Exact Match on the test set for the Setup C and We bold the best result in each section.

	Hindi	German	Spanish	Arabic	Chinese	Vietnamese	English
XLM-RoBERTa$_{base}$	59.1 / 76.9	51 / 94.9	53 / **74.4**	**50 / 92.7**	**44.8** / 60.2	57.2 / 81.6	73.6 / 49.3
Task Adapter (*Pfeiffer*)	58.2 / 84.7	51 / **93**	52.2 / 88.9	49 / 92.8	43.9 / **59.2**	**57.3** / 81	73.4 / 50.7
Task Adapter (*Houlby*)	59.7 / **70.6**	51.1 / 93.4	53.1 / 78.9	48.5 / 87.6	43.6 / 60.1	57.1 / 79.9	73.6 / 49.7
Language Adapter	**60.4** / 74.7	**52.5** / 104.2	**54.6** / 75.9	49.2 / 93.8	44.3 / 59.7	56.6 / **76.4**	**73.8** / 49.4
Task + Language Adapter	59.5 / 82.8	51.6 / 97.4	53 / 76.9	49.8 / 93.7	44.1 / 59.4	57.2 / 85.2	73.7 / **46.5**
MAD-X (Multi-Task Adapter)	59.7 / 72.6	48.6 / 95.5	50.3 / 88.7	42.9 / 107	42.4 / 60.9	53.7 / 88.4	-

Table 5: Jaccard and Word error rate (WER) on the test set for Setup A, B, C, and D

	Multi-Task Adapter (Task + Source Language + Target Language)
Hindi	65.24 / 48.91
German	60.42 / 43.35
Spanish	65.82 / 44.2
Arabic	50.12 / 31.33
Chinese	42.87 / 41.86
Vietnamese	64.48 / 44.22
English	-

Table 6: F1 score and Exact Match on the test set for the Setup D of Multi-Task adapter

style. The general poor performance in the Arabic language could be due to a variation in scripting technique. This also demonstrates that, regardless of the downstream task, the language structure has a significant impact on overall performance.

The Chinese language has a symbolic language structure and can be written in a variety of forms (left-to-right, or vertically top-to-bottom). The degraded findings in Chinese compared to other low-resource languages are most likely due to the language's writing flexibility.

Acknowledgments

The PARAM Shavak HPC computer facility is used for some of our experiments. We are grateful to the Gujarat Council of Science and Technology (GUJCOST) for providing this facility to the institution so that deep learning studies are being carried out effectively.

5 Conclusions

We have investigated the efficacy of cascading adapters with transformer models to leverage high-resource language to improve the performance of low-resource languages on the question answering task. We trained four variants of adapter combinations for - Hindi, Arabic, German, Spanish, English, Vietnamese, and Simplified Chinese languages. We demonstrated that by using the transformer model with the multi-task adapters, the performance can be improved for the downstream task. Our results and analysis provide new insights into the generalization abilities of multilingual models for cross-lingual transfer on question answering tasks.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. 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)*, pages 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. RobBERT: a Dutch RoBERTa-based Language Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3255–3265, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of

deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

Rudra Murthy, Anoop Kunchukuttan, and Pushpak Bhattacharyya. 2019. Addressing word-order divergence in multilingual neural machine translation for extremely low resource languages. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3868–3873, Minneapolis, Minnesota. Association for Computational Linguistics.

HA Pandya and BS Bhatt. 2021. Question answering survey: Directions, challenges, datasets, evaluation matrices. *Journal of Xidian University*, 15(4):152–168.

Youngja Park, Siddharth Patwardhan, Karthik Visweswariah, and Stephen Gates. 2008. An empirical analysis of word error rate and keyword error rate. pages 2070–2073.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020b. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Tulika Saha, Dhawal Gupta, Sriparna Saha, and Pushpak Bhattacharyya. 2021. A unified dialogue management strategy for multi-intent dialogue conversations in multiple languages. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 20(6).

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.