Towards More Knowledgeable Reasoning: Experiments on Natural Language Inference

Ziqiao Ma
Undergraduate, EECS
University of Michigan
marstin@umich.edu

Xueming Xu
Undergraduate, EECS
University of Michigan
xueming@umich.edu

Qingyi Chen
Undergraduate, EECS
University of Michigan
chenqy@umich.edu

Abstract

Natural Language Inference (NLI) has been one of the recent focuses in the NLP community. NLI tasks generally require machines to possess a strong reasoning ability and a broad understanding of words, and even the world. This project aims at approaching three categories of tasks in NLI: question answering, textual entailment, and plausibility inference. We conclude that with the input of extra knowledge from other datasets or knowledge graphs, the performances of baseline pre-trained models are improved to different extent. By conducting trials on various models and comparing between them, we learnt and summarized the strategies of building Natural Natural Inference models that worked best for us.

1 Introduction

1.1 Natural Language Inference

The process of reasoning and inference is crucial for both human and artificial intelligence when addressing natural language sources. Specifically, Natural language inference (NLI) can be formulated as a family of text classification problems, where the fundamental task is to classify the relationship between sentences, including entailment, contradiction and more (Chen et al., 2020). Such tasks particularly challenge machines' capability of capturing underlying information in text and external knowledge about language and the world.

Although reasoning beyond explicitly expressed is trivial to human, the task remains challenging for machines, thus compiled knowledge resources are introduced in support. Storks et al. (2020) summarized three types of knowledge resources: linguistic knowledge, common knowledge, and commonsense knowledge, corresponding to the linguistic knowledge, explicit facts and implicit common senses.

1.2 Language Model Pre-training

The Transformer proposed by (Vaswani et al., 2017) was a major breakthrough in translation quality, and it provided an alternative model architecture for a wide spectrum of NLP tasks. It has been so influential that the majority of the state-of-the-art approaches for NLI depend on later variations of pre-trained linguistic models (Zhou et al., 2019).

Depending on their high-level strategies, these models fall into one of the following categories: autoregressive models, autoencoding models, sequence-to-sequence models, multimodal models and retrieval-based models (Babić et al., 2020). Particularly, autoencoding models are pre-trained by corrupting the original sentence and then then reconstructing it, including BERT (Devlin et al., 2019), ALBERT (Lan et al., 2020) and RoBERTa (Liu et al., 2019). Meanwhile, autoregressive models are pre-trained by guessing the next token given previous ones, including GPT (Radford et al., 2018), GPT2 (Radford et al., 2019) and XLNet (Yang et al., 2020).

Pre-trained language models can be fine-tuned for text classification tasks. The language models are trained in a general corpus, with different data distribution from the target domain. Sun et al. (2020) summarized 3 approaches for further pre-training:

- **Within-task pre-training**: pre-train the model on the training data of a target task;
- **In-domain pre-training**: pre-train the model on data of similar distribution;
- **Cross-domain pre-training**: pre-train the model on data of possibly different domains to a target task.

1.3 Commonsense Knowledge

Beyond pre-training, the community has a continuous effort in incorporating external knowledge,

Benchmark	CommonsenseQA	ConvEnt	EAT
Task Type	Question Answering	Textual Entailment	Plausible Inference
Training size	9741	442	887
Validation size	1221	78	157

Table 1: Benchmarks for Natural Language Inference Experiments. For ConvEnt and EAT dataset, the training and validation sets are split randomly with a ratio of 85%:15%.

especially common and commonsense knowledge. Such knowledge are produced in various ways, including generating from human-annotated evidence like WikiNLI (Chen et al., 2020), mining from pretrained models (Davison et al., 2019) and extracting evidence from a knowledge sources.

Knowledge sources of different natural structures are available, including graph-structured knowledge like ConceptNet (Speer et al., 2018) and unstructured/semi-structured knowledge like Wikipedia plain texts (Ryu et al., 2014). Figure 1 shows an example from the CommonsenseQA dataset (Talmor et al., 2019) which requires multiple external knowledge to make the correct prediction. In this example, evidence from ConceptNet helps to rule out choices (B,D,E) and Wikipedia text evidence helps rule out choices (A,B,D). With the knowledge input from both knowledge sources, machines can derive the correct answer C.

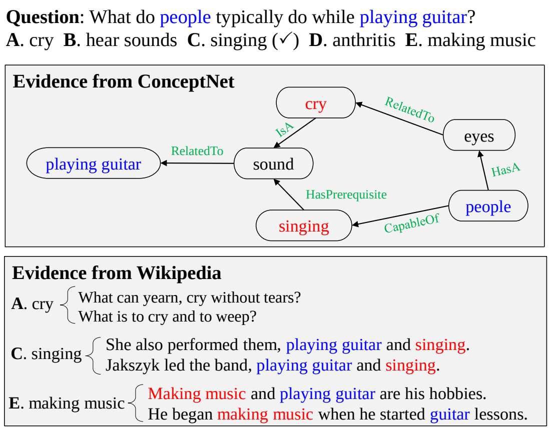

Figure 1: A question answering example from the CommonsenseQA dataset. Making the correct answer requires external knowledge from both ConceptNet and Wikipedia. Words in blue are the concepts in the question. Words in green are the relations from ConceptNet. Words in red are the choices picked up by evidence (Lv et al., 2020).

Particularly, graph-structured knowledge is proved to be powerful in many application, because of its ability to represent words as individual nodes and relationships between words as edges. To handle graph information, recent years have seen a series of work using graph neural networks to introduce knowledge graphs for NLI tasks. Inspired by Lv et al. (2020) and Song et al. (2020), we propose to use graph convolutional networks to extract knowledge graphs collected evidence from heterogeneous external knowledge sources, and develop a graph-based reasoning framework to provide extracted knowledge to NLI models.

2 Tasks and Benchmarks

To study the capability of graph attention based reasoning framework to address general NLI tasks, we perform experiments on three different types of NLI problems: Question Answering, Textual Entailment and Plausible Inference. One benchmark dataset is chosen for each task as is listed in Table 1, each requiring the model to perform causal reasoning upon comprehensive commonsense.

CommonsenseQA CommonsenseQA is a question answering benchmark (Talmor et al., 2019). It presents a natural language question Q of m tokens $\{q_1, q_2, \cdots, q_m\}$ and 5 choices $\{a_1, a_2, \cdots, a_5\}$ labeled with $\{A, B, \cdots, E\}$.

ConvEnt ConvEnt (Conversation Entailment) is a textual entailment task studied by Zhang and Chai, 2010). It features a conversation Q composed of n sequences of natural language texts $\{t_1, t_2, \cdots, t_n\}$ as the premise and an interpretation sentence h as the hypothesis. The task is to identify if the hypothesis h is entailed in the given dialogue.

EAT EAT (Everyday Actions in Text) is a plausible inference benchmark from the *SLED* group. The dataset consists of a sequence of events represented by natural language texts $\{t_1, t_2, \cdots, t_5\}$. The model aims to identify whether the story is plausible and if not, specify at which event the story becomes implausible.

3 Computational Models

In this report, we will be comparing methods that utilize knowledge sources in different ways. To make comparisons, we classify the models into 3 groups, as is shown below and in Table 2.

- **Group 1**: within-task tuned models;
- **Group 2**: in/cross-domain tuned models;
- **Group 3**: graph based models.

BenchmMarks	Group 1	Group 2	Group 3
CommonsenseQA	Y	N	Y
ConvEnt	Y	Y	N
EAT	Y	Y	N

Table 2: All of the 3 benchmarks are experimented on Group 1 models for baseline comparison. Since CommonsenseQA is a large dataset while ConvEnt and EAT are small, there is no need to introduce data of similar domains for CommonsenseQA, while graph models cannot generate representations on very small datasets. Therefore, only ConvEnt and EAT benchmarks are experimented on Group 2 models, and the Group 3 model is only applied to CommonsenseQA.

3.1 Within-task Tuned Models

The within-task tuned models are pre-trained language models that is directly fine-tuned on the target training data and make predictions directly on the validation dataset without external knowledge sources involved. We chose 2 autoencoding models: BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) and 1 autoregressive model: XLNet (Yang et al., 2020) for experiments for all of the 3 benchmarks. These models are expected to provide a comparison baseline for task performances when no knowledge outside of the target dataset is involved.

Type	Models
Autoencoding	BERT, RoBERTa
Autoregressive	XLNet

Table 3: A table of within-task pre-trained models for experiments.

3.2 In/Cross-domain Tuned Models

The ConvEnt and EAT benchmarks are too small in terms of training set size, and the training loss stops updating quickly after a few epochs. Therefore, it can be expected before experiment that within-task tuning would perform poorly on these benchmarks.

Recall from Section 1.2, in-domain pre-trained models are tuned on data of similar distribution, and cross-domain pre-trained models are tuned on data of possibly different domains to a target task. The general procedure is to first fine-tune our model on these knowledge source datasets, and then tune the model on the target benchmark. To set up in/cross-domain fine-tuning, we chose 3 extra datasets for knowledge sources in each of the benchmark types.

- **Question Answering**: PIQA, created by Bisk et al. (2019).
- **Textual Entailment**: MultiNLI, created by Williams et al. (2018)
- **Plausibility Inference**: SWAG, created by Zellers et al. (2018)

We expect that datasets of the same benchmark types should share a more similar domain distribution, while datasets of different benchmark types should share a different domain distribution, as is summarized in Table 4.

Benchmarks	In-domain	Cross-domain
ConvEnt	MultiNLI	PIQA, SWAG
EAT	SWAG	PIQA, MultiNLI

Table 4: For ConvEnt, models runed on MultiNLI are in-domain, while models tuned on SWAG and PIQA are cross-domain. Similarly for EAT, models runed on SWAG are in-domain, while models tuned on MultiNLI and PIQA are cross-domain.

3.3 Graph Based Models

The community has started to use graph neural networks (GNNs) to introduce external knowledge to address many NLI tasks. Graph-based networks are models that extract and learning knowledge representations from graph-structured knowledge sources and make inferences upon these external evidences. In the report, we used the graph-based reasoning model by Lv et al. (2020) to experiment on CommonsenseQA dataset.

The KGAnet proposed by Song et al. (2020) address the Textual Entailment problem and perform experiment on SNLI (Bowman et al., 2015). The module applies a cross-attention mechanism in extracting prediction, and is proved to outperform traditional Graph Attention Network (GAT) (Veličković et al., 2018). However, the inference module of this model is not graph-based, so we applied the graph-based reasoning model for experiments.

The graph-based reasoning model proposed by Lv et al. (2020) is an adaptation of XLNet (Yang et al., 2020). One major contribution of the work is that they are the first to propose a model that leverage evidence from multiple knowledge sources. In the experiment, ConceptNet and Wikipedia Plain Text are preprocessed into knowledge graphs.

The graph-based reasoning module, as is represented in Figure 2, consists of a graph-based contextual representation learning module and a graph-based inference module.

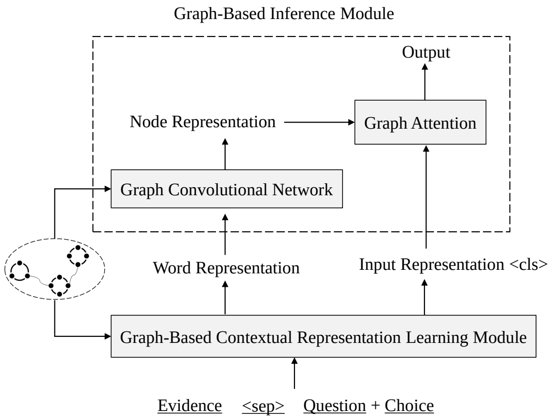

Figure 2: An overview of our proposed graph-based reasoning model (Lv et al., 2020)

The graph-based contextual representation learning module is built upon XLNet (Yang et al., 2020). The module assigns a closer distance of those related works in different evidence sentences by using graph information. Algorithmically, Topology Sort Algorithm is applied to re-order the input evidence according to the constructed knowledge graphs.

The graph-based inference module tries to aggregate evidence at the graph-level for predictions. Specifically, a Graph Convolutional Network (GCN) (Kipf and Welling, 2016) is used to retrieve the node representation, and a graph attention layer is applied for prediction.

4 Experimental Results

This section delivers our experimental results for each benchmarks[1]. The majority of the codes are developed in the `HuggingFace` framework (Wolf et al., 2020).

4.1 CommonsenseQA

The experiment results are listed in Table 5.

[1]The source codes are available at https://github.com/Mars-tin/commonsense-for-inference

Group	Model	Val Acc (%)
Random	Random	20.0
Group 1	BERT-base	56.6
	BERT-large	61.7
	XLNet-base	46.9
	XLNet-large	62.7
	RoBERTa-base	67.2
	RoBERTa-large	**77.4**
Group 3	Graph Based (our)	73.0
	Graph Based (official)	79.3

Table 5: The validation accuracy obtained for each model tested. All the values are the best outcome after hyperparamter tuning, including learning rate, decay rate, etc.

Data Preprocessing The CommonsenseQA task is formulated as a multiple choice problem, a subset of text classification problem.

The dataset is in the form of a natural language question Q of m words $\{q_1, q_2, \cdots, q_m\}$ and 5 choices $\{a_1, a_2, \cdots, a_5\}$ labeled with $\{A, B, \cdots, E\}$. For each question, five inputs are formulated by concatenating the question and each answer. We also signify the relation of question and answer in the input by adding a "Q" before the question and an "A" before the answer so that the input will be formulated as $\{$Q: $q_1\ q_2 \cdots q_m$ A: a_i $\}$. The formal input is tokenized and composed of special tokens, such as separation tokens between the question and the answer and padding tokens following the original sentence.

Best Performing Model The best performing model among all experiments is the fine-tuning RoBERTa-large model (Liu et al., 2019), implemented in `fairseq` framework (Ott et al., 2019). The model ends up with a validation accuracy of 77.4%.

The model was tested for several sets of hyperparameters, the best result came from the model trained in 10 epochs, using an AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta = (0.9, 0.98)$, $\varepsilon = 10^{-6}$ and learning rate of 10^{-5}. The dropout rate is set to 0.1.

5 Conversation Entailment

The experiment results are listed in Table 6.

Data Preprocessing The ConvEnt task is formulated as a binary sequence classification problem, a subset of text classification problem.

Group	Model	Val Acc (%)
Random	Random	50.0
Group 1	BERT-base	54.8
	BERT-large	57.6
	XLNet-base	54.8
	XLNet-large	54.8
	RoBERTa-base	54.8
	RoBERTa-large	60.9
Group 2	On PIQA	63.1
	On SWAG	65.0
	On MultiNLI	**66.3**

Table 6: The validation accuracy obtained for each model tested. All the values are the best outcome after hyperparamter tuning, including learning rate, decay rate, etc. All the baseline pretrained models for group 2 are RoBERTa-large. Many models ended up with 54.8% because all the predictions are 1, and the model did not learn from the data due to the size.

The ConvEnt dataset consists of a conversation Q composed of n sequences of natural language texts $s_1 = \{t_{1,1}, t_{1,2}, \cdots, t_{1,m_1}\}$, $\cdots$, $s_n = \{t_{n,1}, t_{n,2}, \cdots, t_{n,m_n}\}$ as the premise and an interpretation sentence h as the hypothesis.

To prevent the potential issue in tokenization (for example, both "speakerA" and "speakerB" are tokenized to the same token id), we substitute every appearance of "speakerA" with "Tom" and every appearance of "speakerB" with "Bob", which can be well tokenized to different token id's. We also substitute pronouns such as "I" and "you" to their corresponding subjects to make the reference relation clearer.

To formulate the input, every part of the conversation, as well the hypothesis, is concatenated together as $\{s_1, \cdots, s_2, h\}$. The formal input is tokenized and composed of special tokens, such as separation tokens between the premise and the hypothesis and padding tokens following the original sentence.

Best Performing Model The best performing model among all experiments is the fine-tuning RoBERTa-large model (Liu et al., 2019) on MultiNLI dataset Williams et al. (2018), implemented in `HuggingFace` framework (Wolf et al., 2020). The model ends up with a validation accuracy of 66.3%.

The model was tested for several sets of hyperparameters, the best result came from the model trained in 1 epoch (since the dataset is small), using an AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta = (0.9, 0.98)$, $\varepsilon = 10^{-6}$ and learning rate of 1×10^{-6}.

5.1 EAT

The experiment results are listed in Table 7.

Group	Model	Plausibility Accuracy (%)	Breakpoint F1-score (%)
Random	Random	50.0	20.0
Group 1	BERT-base	54.1	25.1
	BERT-large	56.1	42.6
	XLNet-base	50.3	22.3
	XLNet-large	50.3	22.3
	RoBERTa-base	63.4	56.6
	RoBERTa-large	73.1	64.5
Group 2	On PIQA	64.6	55.2
	On SWAG	75.4	**67.6**
	On MultiNLI	**77.5**	65.3

Table 7: The validation plausibility accuracy and breakpoint F1-score obtained for each model tested. All the values are the best outcome after hyperparamter tuning, including learning rate, decay rate, etc. All the baseline pretrained models for group 2 are RoBERTa-large.

Data Preprocessing The Everyday Actions in Text task is formulated as a multiple choice of a list of binary sequence classification, which can be interpreted as a combination of text classification problems.

The EAT dataset consists of a sequence of n events represented by natural language texts $\{t_1, \cdots, t_n\}$. To examine whether the whole story is plausible, we sequentially compose the input by concatenating consecutive events. For example, a sequence of 4 events $\{t_1, t_2, t_3, t_4\}$ will make up 3 inputs: $\{t_1, t_2\}$, $\{t_1, t_2, t_3\}$, and $\{t_1, t_2, t_3, t_4\}$. This method of splitting the whole story conveniently helps the model to focus on the relation between events and helps us determine where the breakpoint may occur.

The formal input is tokenized and composed of special tokens, such as separation tokens before the final event in each input and padding tokens following the original sentence.

Best Performing Model The best performing model among all experiments is the fine-tuning RoBERTa-large model (Liu et al., 2019), implemented in `HuggingFace` framework (Wolf et al., 2020). The best plausibility accuracy is obtained when the model is first tuned on MultiNLI Williams et al. (2018), while the best breakpoint F1-score is obtained when the model is first tuned on SWAG Zellers et al. (2018). The best accuracy is 77.5% and the best breakpoint F1-score is 67.6%.

The model was tested for several sets of hyper-parameters, the best result came from the model trained in 1 epoch (again, the dataset is small), using an AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta = (0.9, 0.98)$, $\varepsilon = 10^{-6}$ and learning rate of 2×10^{-6}.

6 Discussion

6.1 Feedback

With the experiments, we arrived at some insights in handling NLI tasks.

Insight 1 Autoencoding models outperform autoregressive ones.

The first important insight is that, in general, one would expect autoencoding models like BERT and RoBERTa to do better in text classification tasks than autoregression models like XLNet.

According to our experiment results, the performance of XLNet is usually less satisfying. This is especially the case in ConvEnt and EAT task. In these tasks, the prediction of XLNet based models on binary classification tasks are blown up and the outputs will be all 0 or all 1, leading to the near-random performance. While for the autoencoding models, with proper hyperparameters, the output can be reasonable, though not satisfying.

We believe that this nature can be explained by the origin of these models. the autoencoding models are trained by corrupting one sentence then reconstructing it, thus is naturally suitable for sentence classification or token classification tasks. Meanwhile, autoregressive models are developed in traditional natural language generation tasks, thus are more suitable for text generation tasks.

Insight 2 Graph based models help to improve theoretically but is computationally expensive for practical applications.

Our re-implementation of the graph based model ended up with an accuracy of 73.0%, while the official reported accuracy of the model is 79.3%.

Looking back on the experiment settings of the original paper, it can be found that their experiment was done on 2 P100 GPUs with 50 GB RAM, which is beyond the computational power of Colab. Also, their result was obtained after 40000 epochs of training, which is not affordable for us, as our result was obtained after 500 epochs. Therefore, although graph based models can help to improve the performance of NLI tasks theoretically, in practical perspective, they are less competitive to the user-friendly transformers. In fact, our best result on RoBERTa-large is 77.4%, which is almost equal to the official accuracy.

One of the possible research topic could be developing pre-trained GNN models and graph embeddings for natural language tasks, and this would definitely be powerful for industry.

Insight 3 In-domain tuning are more powerful than cross-domain tuning.

For smaller datasets like ConvEnt and CommonsenseQA, the model does not learn from the dataset and the predictions ended up with all-0 or all-1. In such cases, it is important to apply in-domain or cross-domain training, with knowledge input from other datasets.

According to our experiment results, the best accuracy on ConvEnt was obtained by pre-tuning the RoBERTa-large model on the MultiNLI dataset, which is as well a textual entailment benchmark. Also, the best breakpoint F1-score for EAT is obtained by training the RoBERTa-large model first on 25000 samples from SWAG dataset (the full dataset contains over 70000 samples, but we cannot afford the time, to train on such a large dataset), and SWAG is a plausible inference benchmark.

In general, in-domain tuning are more powerful than cross-domain tuning. We believe that the reason behind is that datasets of the same benchmark types share a closer distribution, thus the knowledge learned in one dataset transform well to the other one.

6.2 Error Analysis

To analyze the cause of error, we inspected a few wrong predicting instances from the best performing model and tried to figure out potential explanations for the mis-predictions.

Table 8, 9, 10 samples a few typical wrong predictions for CommonsenseQA, ConvEnt, and EAT.

CommonsenseQA We think the problem lies in lack of knowledge or insufficient extraction. The questions that are wrongly predicted generally requires a very strong reasoning and understanding on the interrelationships of words, which, is what the graph based model is good at. In fact, the

Question	Choices		Answer	Prediction
James was looking for a good place to buy farmland. Where might he look?	A. midwest C. estate E. Illinois	B. countryside D. farming areas	A	E
What do people typically do while playing guitar?	A. cry C. singing E. making music	B. hear sounds D. arthritis	C	E
Where could you find a toilet that only friends can use?	A. rest area C. stadium E. hopital	B. school D. apartment	D	B

Table 8: Mis-predicted examples in CommonsenseQA benchmark

Conversation	Hypothesis	Answer	Prediction
SpeakerA: I'm, just wrote my resume up because told we might be facing layoff over at Digital and they've never had, well, they've had layoffs recently, but when we got hired here, no, no, never any layoffs, never, never,	SpeakerA thinks there will be layoffs at Digital	Entailment	Non-Entailment
SpeakerB: Well, I like animals, but we don't have any yet. We have a nine month old with another on the way SpeakerA: Uh-huh. SpeakerB: and we thought, well maybe when they're a little bit bigger	SpeakerB thinks that her kids are too small for animals	Entailment	Non-Entailment

Table 9: Mis-predicted examples in Conversation Entailment benchmark

Story	Plausibility (breakpoint)	Prediction (breakpoint)
Ann stepped into the garage. Ann turned on the washing machine. Ann put the detergent in the washing machine. Ann put the shorts in the washing machine. Ann walked out of the bathroom.	Implausible (4)	Plausible(-1)
Tom took cake from fridge. Tom peeled the orange with knife. Tom throws away his ice cream. Tom ate ice cream with spoon. Tom put cake into oven.	Implausible (3)	Implausible(1)

Table 10: Mis-predicted examples in EAT benchmark

second mis-prediction example is used for demonstration in the graph based model paper.

Conversation Entailment We think the wrong predictions originate from complex or implicit structures in in the conversation as well as vaguely referencing pronouns. For example, the conversation in the second row of Table 9 is wrongly predicted possibly because the pronoun "they" is not well understood by the model.

EAT We think the wrong predictions came from a lack of similar patterns in the fine-tuning dataset. For example, the model is not able to identify the breakpoint 4 in the first row of Table 10, which demands identifying the long-distance dependency between first sentence and the fifth sentence. Yet, the knowledge dataset, MultiNLI, is originally designed for reasoning over two short sentences. Also, our model is prone to sharp topic shift between sentences as indicated by the second row of Table 10.

6.3 Conclusion

In this project, we can come to the conclusion that, **with the input of extra knowledge from other datasets or knowledge graphs, the performances of baseline pre-trained models are improved to different extent**.

References

Karlo Babić, Sanda Martinčić-Ipšić, and Ana Meštrović. 2020. Survey of neural text representation models. *Information*, 11(11):511.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. Piqa: Reasoning about physical commonsense in natural language.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Mingda Chen, Zewei Chu, Karl Stratos, and Kevin Gimpel. 2020. Mining knowledge for natural language inference from wikipedia categories.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. 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 1173–1178, Hong Kong, China. 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.

Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Pum-Mo Ryu, Myung-Gil Jang, and Hyun-Ki Kim. 2014. Open domain question answering using wikipedia-based knowledge model. *Information Processing Management*, 50(5):683 – 692.

Meina Song, Wen Zhao, and Ee Haihong. 2020. Kganet: a knowledge graph attention network for enhancing natural language inference. *Neural Computing and Applications*, 32.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2018. Conceptnet 5.5: An open multilingual graph of general knowledge.

Shane Storks, Qiaozi Gao, and Joyce Y. Chai. 2020. Recent advances in natural language inference: A survey of benchmarks, resources, and approaches.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2020. How to fine-tune bert for text classification?

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2020. Xlnet: Generalized autoregressive pretraining for language understanding.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference.

Chen Zhang and Joyce Chai. 2010. Towards conversation entailment: An empirical investigation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 756–766.

Chen Zhang and Joyce Y. Chai. What do we know about conversation participants: Experiments on conversation entailment.

Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. 2019. Evaluating commonsense in pretrained language models. *CoRR*, abs/1911.11931.