CS 162 Project Final Report

Jonathan Carlson and **Thomas Kamm** and **Chris Lok** and **Matthew Craig**
University of California, Los Angeles
{jonathancarlson, tkamm, chrislok, mcraig90505}@g.ucla.edu

1 Introduction

Commonsense reasoning is challenging for language models because it requires the application of basic world knowledge, logical inference, and an understanding of causality. This paper evaluates different models and tests them on common sense datasets, such as COM2SENSE (Singh et al., 2021) and SemEval-2020 (Wang et al., 2020); the goal is to use these data sets to finetune pretrained large language models (via transfer learning) such as RoBERTa, DeBERTa, and DeBERTa V3 on the specific task of performing common sense reasoning between similar statements.

This final report details our experiments to finetune and improve our model performance on the downstream COM2SENSE task. Our experiments demonstrate that the best finetuned base (∼100M parameters) pretrained language models achieve ∼70% accuracy and ∼52% pairwise accuracy on the COM2SENSE test and dev sets, a significant improvement over the baseline. Our code is available in Appendix A.

1.1 Com2Sense Dataset

The COM2SENSE dataset is used as an evaluation benchmark for commonsense reasoning in NLP models. The dataset consists of pairs of complementary statements with minimal differences except for the relevant comparison. For example:

> "If you are baking two pies, you should double your recipe"

And the complementary statement:

> "If you are baking two pies, you should triple your recipe"

Because of the complementary nature of the COM2SENSE dataset, we include the pairwise accuracy evaluation defined in Singh et al. 2021. Pairwise accuracy is correct only if the prediction is correct on both samples in the complementary pair. This metric tests whether the model is really learning the underlying reasoning behind the related statements, or is making lucky guesses.

2 Methods

We trained and evaluated each of the models on Google Compute Engine using virtual machines (VMs) equipped with graphics processing units (GPUs). All models were trained on 1 Nvidia P100 GPU, which was sufficient for all three of the models used but proved to be insufficient for larger models (e.g. DeBERTa-V3-large) that prevented their use. Larger models required more GPU memory, which restricted training to small batch sizes and greatly increased training time. With limitations of both time and GCP credit, we opted to use the smaller base models and have the flexibility to use larger batch sizes, which allowed for more trials trained on larger data sets. Our GCP credit was not sufficient to use multiple GPUs to train larger models.

We considered the following pretrained masked language models. The same learning rate was used for all of the models (the given default of 1e-5), but the batch size and number of epochs were varied.

RoBERTa The RoBERTa-base (125M) model introduced in (Liu et al., 2019) built on the BERT-base architecture from (Devlin et al., 2019)

DeBERTa DeBERTa (He et al., 2021b) improves upon RoBERTa and BERT by using a disentangled attention mechanism where word's representation is embedded into separate content and position vectors. We use the DeBERTa-base (100M) variant in our experiments.

DeBERTaV3 DeBERTaV3 (He et al., 2021a) builds on DeBERTa by pre-training with an ELECTRA-style objective (ELECTRA (Clark et al., 2020) is a method of pre-training text en-

coders as discriminators instead of generators) plus gradient-disentangled embedding sharing which significantly improves the model efficiency. We use the DeBERTa-V3-base (86M) variant in our experiments.

2.1 Datasets

Our main strategy in improving model performance was to finetune a large language model on multiple commonsense reasoning datasets before finetuning and evaluating the performance on COM2SENSE. We hypothesized that additional finetuning with commonsense reasoning objectives would increase the performance of the model on COM2SENSE.

We use HuggingFace's `AutoTokenizer` (Wolf et al., 2020) with the "bert-base-uncased" tokenizer to tokenize both datasets into batches with `input_ids`, `attention_mask`, `labels` (if provided), and `token_type_ids` (if present).

We experimented with finetuning on the following datasets:

SemEval-2020 Task 4 Dataset The SemEval-2020 Task 4 dataset (Wang et al., 2020) tests whether a system can correctly tell natural language statements that make sense to humans apart from ones that do not make sense, and to explain the reasoning. For our finetuning objective, we only utilize the complementary sensical and nonsensical statements by making True/False predictions similar to the COM2SENSE objective. We ignore the additional correct and incorrect reasons provided for each training instance.

WinoGrande Dataset The WindoGrande dataset (Sakaguchi et al., 2019) contains 44k problems inspired by the Winograd Scheme Challenge design, which is a set of pronoun resolution problems that aim to test robustness in commonsense reasoning. Each instance of the WinoGrande dataset consists of a question ID, an input statement with a blank to be filled, and two answer options to fill the statement. For our finetuning objective, we substitute each possible answer into the blank and assign the corresponding labels to the resulting sensical and nonsensical statements.

CommonsenseQA 2.0 Dataset CommonsenseQA 2.0 (Talmor et al., 2021) is a challenging dataset that contains yes/no questions about everyday commonsense knowledge, each with a correct answer. We utilize the questions and labels directly for our finetuning objective.

Hellaswag Dataset The HellaSwag dataset (Zellers et al., 2019) consists of 70k commonsense reasoning problems on everyday events that are easy for humans but difficult for machines to correctly solve. Each data instance in HellaSwag contains an ID, a label indicating its topic, a context, and four possible endings for that context, only one of which is a likely option to a human and the rest appearing nonsensical. For our finetuning objective, we append each possible sentence ending to the context and assign the appropriate label according to whether the sentence continuation is sensible or not.

2.2 Model Evaluation

We evaluated our models on their F1 score, loss, accuracy, and pairwise accuracy, with the most importance given to accuracy and pairwise accuracy. The models with the best performance on the respective dev splits of each dataset were chosen for further finetuning on other datasets. Once models had been finetuned on the COM2SENSE train dataset, then the best performing model on the dev split was used to generate the $com2sense\_predictions.txt$ file for evaluating test performance.

3 Experimental Setup

The experiments are designed to meet the following objectives: 1) evaluate the effect of transfer learning across commonsense reasoning datasets; 2) analyze performance on the problem domains and scenarios present in the Com2Sense dataset; and (3) benchmark the performance of state-of-the-art large language models on the commonsense reasoning task.

We performed the following trials. After each trial, the model was finetuned on the Com2Sense train set (to adapt to the to evaluation task) and evaluated using the accuracy, pairwise accuracy, and F1 metrics on the COM2SENSE dev set.

Trial 1 For trial 1 we used DeBERTa-base finetuned on the COM2SENSE train dataset to establish a baseline performance. A batch size of 8 and 4 gradient accumulation steps were used.

Trial 2 For trial 2 we used RoBERTa-base finetuned on the COM2SENSE train dataset to establish a baseline performance. A batch size of 16 and 4 gradient accumulation steps were used.

Trial 3 For trial 3 we used DeBERTa-base finetuned on the SemEval-2020 dataset before finetun-

ing and evaluating on the COM2SENSE dataset. A batch size of 16 and 4 gradient accumulation steps were used.

Trial 4 For trial 4 we used the DeBERTa-base model from trial 3 (finetuned on SemEval-2020) and further finetuned the model on the XL sized WinoGrande dataset before evaluation on COM2SENSE. A batch size of 32 and 4 gradient accumulation steps were used. The dataset took over 30 hours to train on due to the large number of examples.

Trial 5 For trial 5 we used DeBERTa-V3-base finetuned on the CommonsenseQA 2.0 dataset. A batch size of 32 and 4 gradient accumulation steps were used.

Trial 6 For trial 6 we used the finetuned DeBERTa-V3-base model from trial 5 and further finetuned the model on the SemEval-2020 dataset. A batch size of 32 and 0 gradient accumulation steps were used.

Trial 7 For trial 7 we used the finetuned model from trial 6 and finetuned it using the WinoGrande dataset. A batch size of 32 and 0 gradient accumulation steps were used.

Trial 8 For trial 8 we used the finetuned model from trial 7 and finetuned it using the HellaSwag dataset before evaluating the final performance on COM2SENSE. A batch size of 32 and 0 gradient accumulation steps were used.

4 Results and Analysis

Trial	Accuracy	Pairwise	F1	Loss
1	0.56	0.31	0.56	3.56
2	0.56	0.28	0.56	2.95
3	0.61	0.35	0.61	3.73
4	0.63	0.40	0.63	3.65
5	0.69	0.48	0.69	3.12
6	0.70	0.49	0.7	2.55
7	**0.71**	**0.52**	**0.72**	3.12
8	0.71	0.51	0.71	3.21

Table 1: Dev set accuracy, F1-score, and loss for all trials, evaluated on COM2SENSE.

4.1 Results

Table 1 summarizes the performance of all of our experimental trials on the dev set. Our baseline performances for DeBERTa-base and RoBERTa-base in Trials 1 and 2 mirror the baselines established for DeBERTa-large and RoBERTa-large in Singh et al. 2021, but with performance accordingly scaled back in proportion to the smaller base size of each model. RoBERTa-base suffers a much smaller performance penalty compared to its large variant than does DeBERTa-base, which may be due to the relatively smaller decrease in the number of learnable parameters.

The results indicate that finetuning on multiple commonsense datasets increases model performance on all metrics. However, the marginal effect of additional finetuning decreases with the number of datasets. The DeBERTa-V3-base model plateaus quickly around 71% standard accuracy and 52% pairwise accuracy after finetuning on WinoFGrande, CommonsenseQA 2.0, and SemEval-2020 in trials 5-7. The additional tuning on the HellaSwag dataset in trial 8 fails to increase performance further, and actually results in a small decrease in pairwise accuracy and F1-score. The same effect can be observed with the DeBERTa-base model in trials 2-4. Finetuning on SemEval-2020 in trial 3 provides a substantial boost to performance over the baseline in trial 2. The additional tuning on WinoGrande in trial 4 also increases performance, but by a smaller amount.

The final observation is that DeBERTa-V3 significantly outperforms DeBERTa, even though the base model has fewer learnable parameters. This can likely be attributed to the improved pre-training objective and improved model efficiency. Despite having the fewest learnable parameters of the three models tested, DeBERTa-V3 has the best performance by a significant margin after finetuning.

4.2 Com2Sense Analysis

We analyzed the performance of our finetuned model from Trial 7 on the domains and scenarios present in the COM2SENSE dataset. Our analysis code is available with our source code in Appendix A.

In Figure 1 we present the breakdown of results (using standard accuracy) across combinations of domain, scenario and numeracy. We observe that, compared to the DeBERTa-large results in Singh et al. 2021, our finetuned DeBERTa-V3-base model performs much better on examples involving numeracy, even exceeding non-numeracy performance on the physical and social domains.

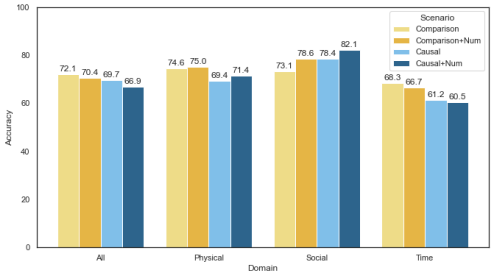

Figure 1: Model performance breakdowns across domains and scenarios. "+Num" denotes examples with numeracy involved.

Performance on the temporal domain is disproportionately worse compared to the other two domains. In particular, performance on causal scenarios in the temporal domain is worse than the DeBERTa-large results in Singh et al. 2021, although performance is significantly better across every other combination of domain and scenario.

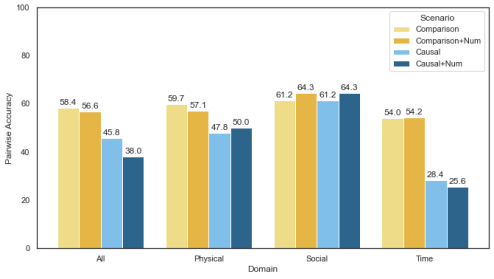

Figure 2: Model performance breakdowns across domains and scenarios using pairwise accuracy metrics.

In Figure 2 we present the pairwise accuracy results across domains and scenarios. We observe that pairwise accuracy in the temporal domain for causal scenarios suffers comparatively more than the other combinations of domain and scenario, which is responsible for a significant decrease in overall pairwise accuracy relative to standard accuracy. It appears that NLP models struggle to make commonsense causal judgements involving time, and are much better at reasoning in the physical and social domains. It is possible that this disparity could be addressed by addressing finetuning specifically on temporal examples.

5 Conclusion

Our best performing model on the COM2SENSE dev set was the Trial 7 model, which used a pretrained DeBERTa-V3-base model finetuned successively on CommonsenseQA 2.0, SemEval-2020, and WinoGrande. The model achieved a standard accuracy of 71% and a pairwise accuracy of 52%, which is performance comparable to the best models from Singh et al. 2021 without finetuning. The model achieved a similar performance on the COM2SENSE test set, with 70% standard accuracy and 52% pairwise accuracy. We observed diminishing performance improvements on the COM2SENSE dev set with each subsequent finetuning, with results plateauing after three datasets and even decreasing in performance on the fourth.

For future work, we hypothesize that using the DeBERTa-V3-large model (or similar large variants of other models) would result in additional performance improvement, in line with the baseline performance reported in Singh et al. 2021. We expect that a similar finetuning approach like we explored could result in double-digit gains in standard and pairwise accuracy over the baseline pretrained model. Additional training epochs could also improve model performance, but we were constrained by time.

Although our best models were able to achieve $>50\%$ pairwise accuracy, we were constrained by both time and GCP credit from creating a baseline model for each finetuning dataset to be able to judge the impact of each individual dataset on model performance. That leaves us with just the marginal performance benefits observed from subsequent finetuning to report the impact of each dataset.

References

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning 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.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale.

Shikhar Singh, Nuan Wen, Yu Hou, Pegah Alipoormolabashi, Te-Lin Wu, Xuezhe Ma, and Nanyun Peng. 2021. Com2sense: A commonsense reasoning benchmark with complementary sentences. *arXiv preprint arXiv:2106.00969*.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandrasekhar Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. 2021. Commonsenseqa 2.0: Exposing the limits of ai through gamification. *ArXiv*, abs/2201.05320.

Cunxiang Wang, Shuailong Liang, Yili Jin, Yilong Wang, Xiaodan Zhu, and Yue Zhang. 2020. Semeval-2020 task 4: Commonsense validation and explanation. *arXiv preprint arXiv:2007.00236*.

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. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Appendix A

Source code available on GitHub: `https://github.com/0x65-e/CS162`