Counterfactual Statement Identification

Yixin Luo
Fudan University
22307110294

Lei Tao
Fudan University
22307110418

Xiaolin Zhang
Fudan University
22307130352

Abstract

The task of counterfactual detection aims to identify statements that describe potential outcomes resulting from hypothetical or unrealized events. To address this task, we trained multiple language models, including BERT, XLNet, CNN with word embedding, RoBERTa and its variants. To improve training efficiency and reduce the number of trainable parameters, we introduced Low-Rank Adaptation (LoRA) for model fine-tuning.

Given the class imbalance in the training data, we add positive example data by rewriting causal statements by a large language model (ChatGLM). Moreover, we proposed a multi-task learning framework , where we used ChatGLM to annotate additional tense information in the training set. This work has improved the performance of our model. Finally, we integrated the 5 best-performing models through a voting-based ensemble approach.

Our final ensemble model achieved **91.87% Recall and 90.88%F1 score on the official test set**, demonstrating strong effectiveness in identifying counterfactual statements.

1 Introduction

Counterfactual statements describe events that did not occur or could not have occurred, along with the possible outcomes had these events taken place, e.g., "*if kangaroos had no tails, they would topple over*". By imagining alternative possible worlds, humans naturally establish connections between the antecedent (e.g., "*if kangaroos had no tails*") and the consequent (e.g., "*they would fall over*"), thereby forming causal judgments—such as concluding that tails help prevent kangaroos from falling. Humans can leverage prior knowledge to comprehend counterfactual statements and to explore the causal relationships they imply. Although it is impossible to reverse events that have already happened or to realize impossible scenarios in the real world, it is still feasible to reason about the potential consequences of alternative events. Counterfactual thinking is a remarkable human capacity, and many researchers consider it the most sophisticated form of causal reasoning. Even the most advanced artificial intelligence systems currently fall far short of achieving human-like counterfactual reasoning. Therefore, counterfactual reasoning is widely regarded as a key component in enhancing the generalization capabilities of AI systems.

To enhance artificial intelligence's ability in counterfactual reasoning, the SemEval-2020 Task 5 [1] established the Counterfactual Recognition task, providing a standardized evaluation platform for identifying counterfactual sentences. This task offers a benchmark dataset for counterfactual recognition in natural language and includes two subtasks: Recognizing Counterfactual Statements and Detecting Antecedent and Consequent. To achieve better training results, we focused solely on the first subtask, experimenting from multiple perspectives such as data augmentation and multitask learning. It is noteworthy that our training objectives differ from the original task, which only emphasizes most on F1 score. We argue that **recall is more crucial than precision** in counterfactual recognition tasks. Since counterfactual sentences are relatively rare in natural language and usually constitute only a small portion of the dataset, recognition systems should prioritize maximizing the retrieval of these key sentences that carry hypothetical and causal significance. If a large number of genuine counterfactual sentences are missed, the practical value of the system's output will be significantly diminished.

In our work, for better evaluating the model performance, we introduced the following methods:

- **Data Augmentation:** We mitigate the class imbalance by generating 1,518 synthetic counterfactual sentences. Using the Causal News Corpus (CNC) as a source of factual event sentences, we prompt the ChatGLM large language model to rewrite them into counterfactual variants. These new examples are added to the training data.

- **Multi-Task Learning:** Hypothesizing that tense is highly relevant to counterfactuals since many counterfactuals use specific tenses, we introduce a joint learning framework that simultaneously performs counterfactual identification and tense recognition. The auxiliary task predicts one of 12 English verb-tense classes.

- **Parameter-Efficient Tuning with LoRA:** As LoRA has emerged as a popular and efficient fine-

[1] https://aclanthology.org/2020.semeval-1.40/

tuning strategy in recent years, we incorporate it into our framework to evaluate its effect on parameter reduction and training efficiency for counterfactual detection.

- **Ensemble of Models:** We train several models using the augmented data and the multi-task framework, then combine their predictions.

2 Related Work

Early approaches to counterfactual recognition relied on surface features and rule-based patterns. The research on counterfactual statement recognition tasks can be traced back to the linguistic studies conducted by Ferguson(Ferguson and Sanford, 2008). Through presenting subjects with both factual and counterfactual statements, they identified **three hallmark linguistic features of counterfactual constructions**: verb tense (e.g., "had done"), conditional sentence structures (e.g., "if...then..."), and negation forms (e.g., "without").

Recent SemEval-2020 participants achieved the best results by fine-tuning large pre-trained models.(Yang et al., 2020) Almost all top systems used Transformer-based encoders (BERT, RoBERTa, XLNet, etc.) to extract contextual embeddings. For example, Bai and Zhou (Bai and Zhou, 2020) fine-tuned BERT for the binary classification and modified its upper layers for richer semantics. They then combined multiple BERT variants using hard voting (an ensemble) to achieve 86.3% F1 on Subtask 1. Similarly, the Yseop team(Akl et al., 2020) used a cascaded BERT+MLP system and obtained 85.0% F1. Other teams (e.g. IITK-RSA(Rohin Garg, 2020)) built ensembles of transformers and convolutional heads and reached even higher F1 scores (91% on the test split). These results show that Transformer ensembling is effective for this task.

After that, counterfactual sentence generation has mostly been studied in explainable AI and robustness contexts. Balashankar(Sun et al., 2023) describe Counterfactual Data Augmentation (CDA) frameworks that use generative models to produce hypothetical examples, improving classifier robustness. While their focus was on sentiment or QA tasks, the principle is similar: using a generative model to synthesize diverse counterfactual examples can boost model performance.

Building upon the aforementioned baseline models, numerous methods can further enhance prediction accuracy and model robustness, such as multi-task learning, LoRA fine-tuning, data augmentation:

2.1 Multi-task learning (MTL)

Multi-Task Learning (MTL) has evolved significantly since its formalization by Caruana(Caruana, 1997), demonstrating that shared representations improve generalization across related tasks and achieves better performance than single-task learning approaches. Recent advances leverage pre-trained models like BERT for MTL, achieving state-of-the-art results in tasks ranging from sentiment analysis to named entity recognition (Liu et al., 2019). Dynamic task-weighting methods(Chen et al., 2018) further address optimization challenges in unbalanced MTL scenarios.

2.2 Low-Rank Adaptation (LoRA)

Fine-tuning large pretrained language models has become a standard approach for adapting general-purpose models to domain-specific tasks, and Low-Rank Adaptation (LoRA) has emerged as a promising and effective strategy (Hu et al., 2021). In LoRA, instead of updating the entire set of pretrained weights, low-rank matrices are inserted into the model's architecture—typically within the attention or feed-forward layers.

The core idea behind LoRA is based on the observation that the update matrices induced by fine-tuning often exhibit a low intrinsic rank, meaning that they can be approximated effectively using low-rank decomposition. By constraining the learned parameters to a low-rank subspace, LoRA achieves strong performance while introducing only a small number of additional parameters. As a result, LoRA not only enables efficient adaptation of large models but also facilitates easier experimentation and deployment across multiple downstream tasks. Due to its efficiency and minimal impact on model performance, LoRA has gained widespread adoption in practical applications involving large language models such as BERT, RoBERTa, and more recently, transformer-based architectures like LLaMA and T5.

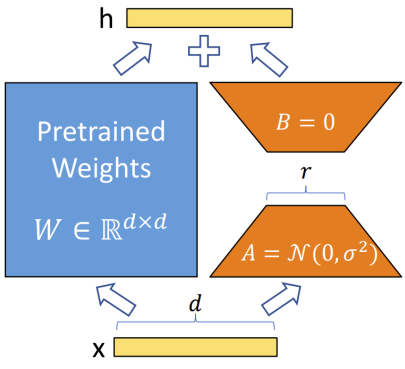

Figure 1: LoRA Fine-tuning Mechanism (Hu et al., 2021)

2.3 Data augmentaion by Counterfactual Generation

Data Augmentation is a technique used to artificially expand a training dataset by applying various transformations or modifications to existing data samples. It helps improve model generalization, prevent overfitting, and enhance robustness, especially when labeled data is limited. Li evaluated the counterfactual statement generation capabilities of large language models(Li et al., 2024). The investigation focused on identifying critical factors affecting text quality in model outputs, while using internal knowledge enhancement and alignment techniques to optimize LLM performance in counterfactual generation.

3 Methodology

3.1 Framework

Inspired by the architectures and results reported in SemEval-2020 Task 5 and its follow-up analyses, we adopt several Transformer-based models as our **baseline backbones**, including **BERT Large**, **RoBERTa Large**, **XLNet Large**, **RoBERTa Large with POS and n-gram features**, as well as **RoBERTa with different classification heads (e.g., CLS, CNN)**. These models have demonstrated strong performance in prior work and serve as the foundation for our extensions. Furthermore, we also trained a CNN classification model using the GloVe word embedding method. However, the effect of this model was poor (Precision= 0.8048, Recall=0.4580, F1=0.5838), so we ultimately did not incorporate it into the integrated model.

Our overall framework consists of three major components: (1) a **data augmentation strategy** to mitigate class imbalance using synthetic counterfactuals generated by a large language model, (2) a **multi-task learning approach** using a training set with newly added temporal labels (annotated by a large language model), and (3) **parameter-efficient tuning** using Low-Rank Adaptation (LoRA) on top of the pre-trained backbones. Figure 2 illustrates the architecture of our complete system.

3.2 Data Augmentation

The original dataset from SemEval-2020 Task 5 is highly imbalanced: only about **11%** of the training instances are labeled as counterfactual, while the remaining **89%** are factual statements. This significant skew leads to biased models that tend to over-predict the majority (factual) class, thereby harming recall on counterfactual examples. To address this, we perform targeted data augmentation by synthesizing additional counterfactual sentences using a large language model.

Specifically, we select causal statements from the **Causal News Corpus (CNC)**(Tan et al., 2022), a dataset composed of causally annotated news articles across diverse domains. These factual sentences are structurally suitable for counterfactual transformation. We then query the **ChatGLM** via its API, using carefully constructed prompts to guide it toward generating plausible counterfactual variants of the original statements.

To improve the lexical and syntactic diversity of the generated sentences, we deliberately design prompts that encourage the use of more complex syntactic structures than relying solely on simpler patterns like "Had...not" or "If...wouldn't". This prompt design aims to reduce the model's sensitivity to common surface-level constructions, thereby enriching the training data with a wider range of counterfactual expressions and improving the model's ability to generalize to varied linguistic forms.

For example, the factual sentence: *"Authorities yesterday said they would scrap the 10.4 billion yuan (HK $12.7 billion) project, the subject of demonstrations by tens of thousands of Shifang residents."* is transformed into the counterfactual statement: *"Were it not for the demonstrations by tens of thousands of Shifang residents, authorities yesterday might not have announced their intention to scrap the 10.4 billion yuan (HK $12.7 billion) project."*

We repeat this process across a wide range of source sentences and retain only those generated outputs that meet basic syntactic and semantic plausibility criteria. In total, we generate **1,518 counterfactual sentences**, which are then appended to the original training set as additional positive examples.

This quantity of augmented data is carefully calibrated: it approximately doubles the number of counterfactual instances in the original dataset, thereby **alleviating the class imbalance** while **avoiding over-reliance on synthetic data**. Excessive augmentation—especially when relying on a single language model—risks introducing stylistic or distributional biases inherent to the generator. These biases may be inadvertently learned by downstream models, impairing generalization to naturally occurring counterfactuals. In contrast, our moderate-scale augmentation strategy ensures that the model continues to learn primarily from the diverse and authentic patterns in the original corpus, while gaining robustness through additional exposure to plausible counterfactual structures. The result is a better balance between bias mitigation and feature diversity.

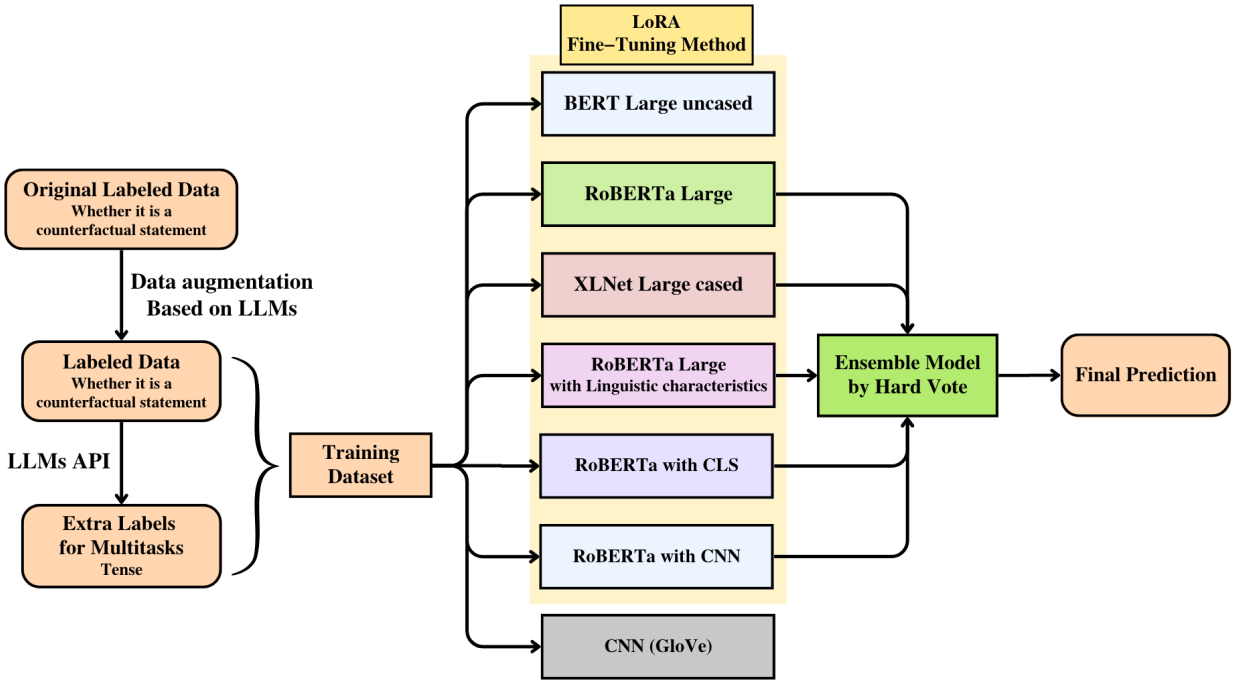

Figure 2: Framework of our task.

3.3 Multi-Task Learning

Counterfactual expressions are often closely tied to **specific tenses**, particularly those used to convey unrealized or hypothetical situations—such as the past perfect or past conditional. Motivated by this linguistic regularity, we introduce an auxiliary task of tense classification into our training framework. By explicitly modeling tense, we aim to help the encoder better attend to morphosyntactic cues that are critical for counterfactual reasoning.

To construct the auxiliary task labels, we apply ChatGLM to recognize the dominant tense of each sentence in the training set. In sentences composed of multiple clauses, we design prompts that encourage the model to focus on the **counterfactual clause**, rather than irrelevant components. For example, given the sentence:

"If he hadn't gone out last night, this wouldn't have happened," she says.

we instruct the model to classify the tense of the counterfactual part (*"If he hadn't gone out last night, this wouldn't have happened"*), rather than the reporting clause (*"she says"*), which conveys no counterfactual meaning.

A subset of sentences—particularly those with ambiguous or fragmented structures—could not be labeled reliably. For these cases, we perform manual annotation to ensure high-quality supervision signals. In total, we annotate tense labels for **14,518 sentences**, covering both the original and augmented datasets.

The tense classification task is formulated as a **12-way classification** problem, following standard English tense taxonomy: simple present, present continuous, present perfect, present perfect continuous, simple past, past continuous, past perfect, past perfect continuous, simple future, future continuous, future perfect, future perfect continuous. Each training instance thus has a pair of labels: one for the main task (binary counterfactual classification) and one for the auxiliary tense classification.

The two tasks are trained jointly in a **multi-task learning (MTL)** framework, using a shared Transformer encoder and two separate classification heads. The loss function is a weighted sum of the two cross-entropy losses.

3.4 LoRA Tuning

In this work, we incorporate Low-Rank Adaptation (LoRA) as a parameter-efficient fine-tuning strategy to enhance the training efficiency of our ensemble model composed of BERT-large, RoBERTa-large, and XLNet-large. LoRA introduces a low-rank matrix factorization method for our language model training that significantly reduces the number of trainable parameters, resulting in faster convergence and lower memory consumption.

In the experiments, we freeze the parameters of the original pre-trained model and introduce a low-rank matrix on its weight matrix to adjust the training of the model. Instead of updating the entire model weights,

only these small, learnable matrices are optimized during training, while the majority of the original model parameters remain frozen. This allows us to maintain the expressive power of the base models while drastically reducing computational overhead.

3.5 Evaluation Metrics

Recognizing Counterfactual Statements is a binary classification problem, so we apply traditional ML metrics for evaluation in our task, including **Accuracy**, **Recall**, and **F1 score**.

$$\text{Recall} = \frac{\#\{\text{True Positives}\}}{\#\{\text{True Positives + False Negatives}\}}$$

We prioritize **recall** over accuracy, for the reason that **missing true counterfactual statements is more detrimental to downstream tasks than incorrectly labeling a few non-counterfactual ones**. In counterfactual recognition, especially given the inherent class imbalance, it is crucial to identify as many true counterfactuals as possible to support reliable causal reasoning, analysis, and subsequent processing. High recall ensures broader coverage of relevant instances, which is essential for the practical utility of the system.

4 Experiment

4.1 Dataset

In our work, we use the official dataset of SemEval-2020 Task 5 [2]. The dataset contains sentences from news articles in the finance, politics, or healthcare domain.

Due to the relative rarity of counterfactual expressions in natural text, comprehensive manual annotation is not feasible in terms of cost and efficiency. Therefore, during the data collection phase, we designed and applied a filtering template that integrates token and part-of-speech (POS) rules to extract candidate sentences potentially containing counterfactuals from the raw corpus. This template set strikes a balance between pattern coverage and linguistic diversity to avoid missing important samples due to overly restrictive filtering conditions.

The candidate samples then proceeded to the manual annotation phase. Each sentence was evaluated by five independent annotators to determine whether it constitutes a counterfactual statement. Counterfactual statements labeled as positive examples were required to have 100% agreement among annotators, while negative examples included samples unanimously identified as non-counterfactual, as well as those with at least 80% agreement. This approach ensured the stability and reliability of the annotation quality.

The final constructed corpus consists of 20,000 annotated sentences, with the training and test sets accounting for 65% and 35%, respectively. The detailed data distribution is shown in Table 1. The amount of data across the three domains is approximately balanced, ensuring representativeness at the register level.

Table 1: Sizes of the training and test set.

Dataset	Counterfact.	Non-counterfact.	Total
Train	1,454	11,546	13,000
Test	738	6,262	7,000
Total	2,192	17,808	20,000

4.2 Experimental setting

We leased GPUs on the AutoDL platform for model training. An RTX 4090 with 24GB of RAM was used for training. For each model, we train 10 epochs.

4.3 Results

4.3.1 Baseline Results

We first evaluate several basic models without any additional augmentation, tuning techniques, or auxiliary tasks. The results are summarized in Table 2(see Appendix).

As shown, all baseline models achieve **competitive performance**, with F1 scores consistently ranging between **86%** and **90%**.

One consistent trend across models is that **recall is always lower than precision**, which implies that models tend to miss a non-negligible portion of actual counterfactual cases. This observation serves as an empirical prior and motivates our subsequent work: we aim to **improve recall while maintaining precision**.

4.3.2 Effect of Data Augmentation

To evaluate the impact of our data augmentation strategy, we re-train the baseline models on the enhanced dataset.Table 3 (see Appendix) summarizes the results of the models trained on the augmented dataset.

Overall, we observe that **data augmentation leads to marginal improvements or slight degradation** in most baseline models. This limited improvement may be due to two factors. First, the baseline models already generalize well from the original training data, and the number of augmented instances accounts for a relatively small portion of the full training set. Second, while synthetic counterfactuals are structurally diverse, they may still differ in style or domain from the original data, slightly affecting model generalization.

[2] https://zenodo.org/records/3932442

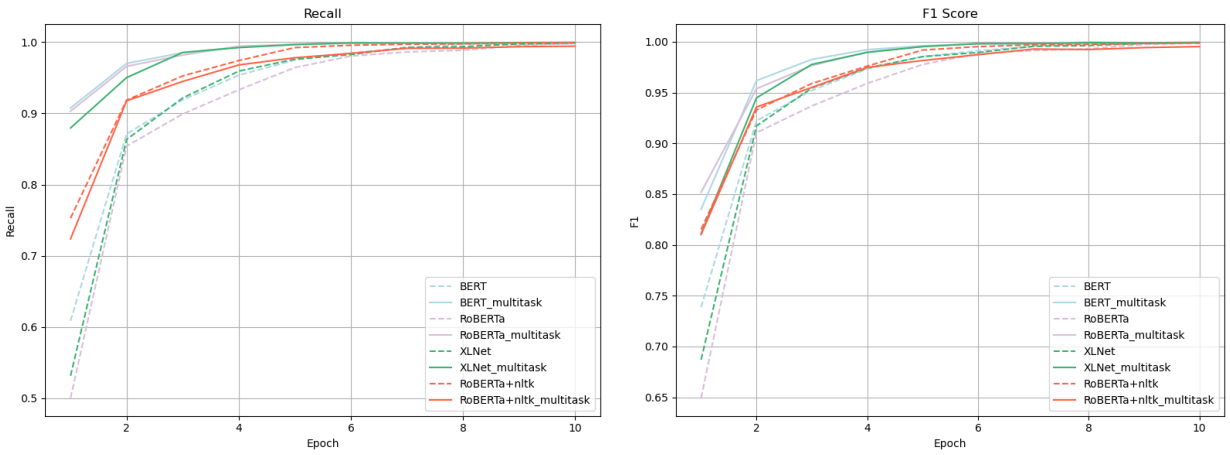

Figure 3: Training set performance of models with and without multi-task learning, plotted across epochs.

However, the **RoBERTa + CLS + augmentation** and **RoBERTa + CNN + augmentation** configurations both demonstrate noticeable performance improvements compared to their non-augmented counterparts. These gains suggest that both classification heads benefit from exposure to additional counterfactual expressions, albeit through different mechanisms:

- For the CLS-based model, which leverages the `[CLS]` token to encode sentence-level semantics, the improvement may stem from better alignment between the augmented data's global counterfactual cues (e.g., "If... had not...") and the model's holistic representation.
- For the CNN-based model, which captures local n-gram features through convolutional filters, the improvement is likely due to the higher diversity of local syntactic patterns introduced by the augmented examples.

These findings reinforce the value of data augmentation, especially when combined with model architectures that emphasize either global sentence semantics (CLS) or local compositional patterns (CNN). While the performance boost is more moderate than initially anticipated, it remains consistent and meaningful across both structural paradigms.

4.3.3 Effect of Multi-Task Learning

We apply the previously mentioned multi-task framework to baseline models and track their training performance across epochs. Figure 3 presents the epoch-wise evolution of recall and F1 score on the training set, comparing the MTL-enhanced models to their single-task counterparts.

As shown in the plots, models trained with multi-task learning exhibit faster convergence and more stable performance in both **recall** and **F1 score**. The auxiliary task appears to guide the model toward learning more generalizable representations that are sensitive to counterfactual constructions.

The benefits of MTL also extend to the test set, where we observe **improvements** in nearly all metrics. Table 4 in the Appendix provides a detailed comparison. Notably, the recall scores show the largest gains—highlighting the auxiliary task's role in helping the model identify subtle counterfactual cues it may have missed in single-task settings.

For example, RoBERTa Large sees its recall rise from 0.8496 (baseline) to 0.8753 (MTL), while its F1 score improves from 0.8843 to 0.8972. These results affirm the intuition that tense recognition serves as a helpful proxy for counterfactual detection, particularly given the strong association between counterfactual meaning and specific verb forms or clause structures.

4.3.4 Parameter-Efficient Fine-tuning with LoRA

After applying LoRA to each backbone model, the number of trainable parameters is reduced to only **0.54% to 1.28%** of the original model size. For example:

- **XLNet Large**: Trainable params / Total params = 1,990,672 / 363,311,122 = **0.5479%**

This drastic reduction in parameter count leads to a significant decrease in training time—The running time of all models with LoRA is reduced by more than **50%** compared with the full fine-tuning model—while also lowering memory usage and disk footprint. These benefits make LoRA especially attractive in low-resource environments or when iterating rapidly across multiple model configurations.

In our experiments, we observe that while LoRA-based models generally match the performance of their fully fine-tuned counterparts, some slight performance drops may occur on specific configurations due to constrained capacity(F1 scores drop within 1%). Nonethe-

less, the trade-off between efficiency and performance is often favorable, especially when deployed at scale or in real-time applications.

4.3.5 Final Ensemble Results

To maximize the strengths of different model architectures and training strategies, we perform a final ensemble by combining multiple high-performing models trained under various settings. The ensemble includes the following five models:

- **RoBERTa Large (MTL)**
- **XLNet Large (MTL)**
- **RoBERTa + CLS (Aug)**
- **RoBERTa + CNN (Aug)**
- **RoBERTa + POS + n-gram (Vanilla)**features.

We use a voting strategy for ensemble prediction, where the final label is determined by the mode of the five model outputs for each test instance.

The ensemble model achieves the following performance on the test set:

- **Precision**: 0.8992
- **Recall**: **0.9187**
- **F1 Score**: **0.9088**

This ensemble surpasses all previous single-model configurations, demonstrating the value of diversity across training paradigms. In particular, the high recall (0.9187) suggests that the combination of multi-task models and augmented data models improves the system's sensitivity to varied counterfactual expressions. Meanwhile, the inclusion of the POS+n-gram variant contributes to structural robustness and surface-level pattern recognition.

Overall, this final ensemble integrates sentence-level semantics (CLS), local syntactic features (CNN, POS), and task-specific representations (tense via MTL), leading to a highly accurate and generalizable counterfactual detection system.

5 Discussion and Conclusion

In this study, we developed an ensemble of powerful language models—including BERT-large, RoBERTa-large, XLNet-large, and several variants incorporating structural and linguistic enhancements—to detect counterfactual statements in text. By leveraging LoRA-based parameter-efficient fine-tuning , we significantly improved training efficiency without compromising model performance. Our final ensemble model achieved Precision = 0.8992 , Recall = 0.9187 , and F1 = 0.9088 , marking a substantial improvement over baseline approaches and demonstrating the effectiveness of our method in the counterfactual detection task.

Despite these promising results, there remain several areas for future improvement. First, while we applied data augmentation using a large language model (ChatGLM), the impact of this strategy was limited and only effective for certain models. Further exploration into more targeted and diverse data augmentation techniques—such as rule-based generation and paraphrasing—could yield better generalization.

Second, our current approach primarily focuses on predictive performance metrics and lacks interpretability analysis. That is, while our models achieve high scores on the test set, we do not investigate why certain predictions are made or how specific linguistic patterns (e.g., tense usage, modal verbs, or causal connectives) influence model decisions. Future work could incorporate attention visualization, feature attribution methods (e.g., LIME or SHAP), or controlled ablation studies to enhance model transparency and provide deeper insights into the nature of counterfactual reasoning.

In conclusion, our system demonstrates strong performance in counterfactual detection, but also highlights the need for further research into data augmentation strategies and model interpretability to better align machine learning systems with human cognitive processes.

References

Hanna Abi Akl, Dominique Mariko, and Estelle Labidurie. 2020. Yseop at semeval-2020 task 5: Cascaded bert language model for counterfactual statement analysis. In *14th International Workshops on Semantic Evaluation, SemEval 2020 - co-located 28th International Conference on Computational Linguistics, COLING 2020, Proceedings*, pages 468–478.

Yang Bai and Xiaobing Zhou. 2020. Byteam at semeval-2020 task 5: Detecting counterfactual statements with bert and ensembles. In *14th International Workshops on Semantic Evaluation, SemEval 2020 - co-located 28th International Conference on Computational Linguistics, COLING 2020, Proceedings*, pages 640–644.

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28(1):41–75.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks.

Heather J. Ferguson and Anthony J. Sanford. 2008. Anomalies in real and counterfactual worlds: An eye-movement investigation. *Journal of Memory and Language*, 58(3):609–626.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Yongqi Li, Mayi Xu, Xin Miao, Shen Zhou, and Tieyun Qian. 2024. Prompting large language models for counterfactual generation: An empirical study.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding.

Shashank Gupta Rohin Garg. 2020. Counterfactuals-nlp. https://github.com/gargrohin/Counterfactuals-NLP?utm$_source = catalyzex.com$.

Hao Sun, Zhexin Zhang, Fei Mi, Yasheng Wang, Wei Liu, Jianwei Cui, Bin Wang, Qun Liu, and Minlie Huang. 2023. MoralDial: A framework to train and evaluate moral dialogue systems via moral discussions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2213–2230, Toronto, Canada. Association for Computational Linguistics.

Fiona Anting Tan, Ali Hürriyetoğlu, Tommaso Caselli, Nelleke Oostdijk, Tadashi Nomoto, Hansi Hettiarachchi, Iqra Ameer, Onur Uca, Farhana Ferdousi Liza, and Tiancheng Hu. 2022. The causal news corpus: Annotating causal relations in event sentences from news. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2298–2310, Marseille, France. European Language Resources Association.

Xiaoyu Yang, Stephen Obadinma, Huasha Zhao, Qiong Zhang, Stan Matwin, and Xiaodan Zhu. 2020. SemEval-2020 task 5: Counterfactual recognition. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 322–335, Barcelona (online). International Committee for Computational Linguistics.

A Appendix

Table 2: Performance of baseline models on the test set. No data augmentation or LoRA is applied.

Model	Precision	Recall	F1 Score
BERT Large	0.8789	0.8455	0.8619
RoBERTa Large	**0.9221**	0.8496	0.8843
XLNet Large	0.9191	0.8618	0.8895
RoBERTa + n-gram	0.9148	**0.8875**	**0.9010**
RoBERTa + CLS	0.8764	0.8715	0.8739
RoBERTa + CNN	0.8791	0.8706	0.8782

Table 3: Model performance on the augmented dataset.

Model	Precision	Recall	F1 Score
BERT Large	0.8838	0.8550	0.8691
RoBERTa Large	**0.9194**	0.8659	**0.8918**
XLNet Large	0.9044	0.8591	0.8812
RoBERTa + n-gram	0.9086	0.8753	0.8916
RoBERTa + CLS	0.9024	0.8767	0.8893
RoBERTa + CNN	0.8760	**0.8997**	0.8877

Table 4: Performance of models with multi-task learning on the test set.

Model	Precision	Recall	F1 Score
BERT Large (MTL)	0.8685	0.8591	0.8638
RoBERTa Large (MTL)	**0.9202**	**0.8753**	**0.8972**
XLNet Large (MTL)	0.9110	0.8740	0.8921
RoBERTa + n-gram (MTL)	0.9196	0.8374	0.8766