CS7.501: Advanced NLP | Project Interim | Monsoon - 24

Team - Semantic Scode

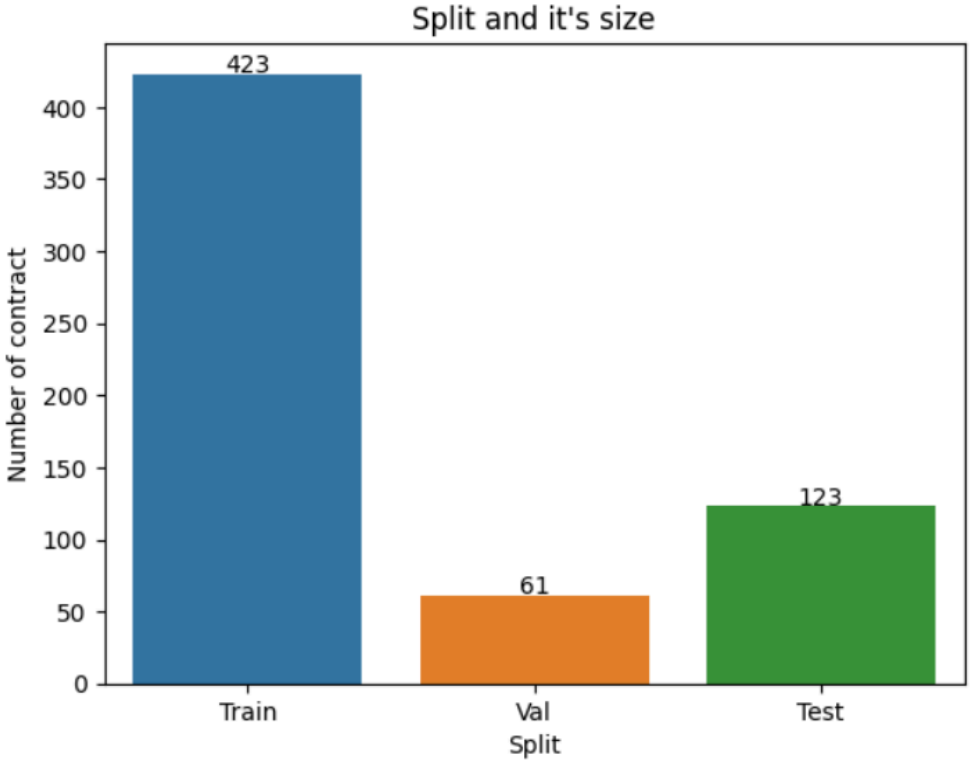
Members

Srivatsa Sinha 2024701003
Shubham Kathiriya 2023201050
Ishita Bansal 2022114004

- Until now, we explored the other different datasets that might be required for a later task or to improve the paper baseline. We did the EDA on the Stanford Dataset of contract NLI and explored the different baselines mentioned in the Contract NLI paper. Exploring the RAG part.

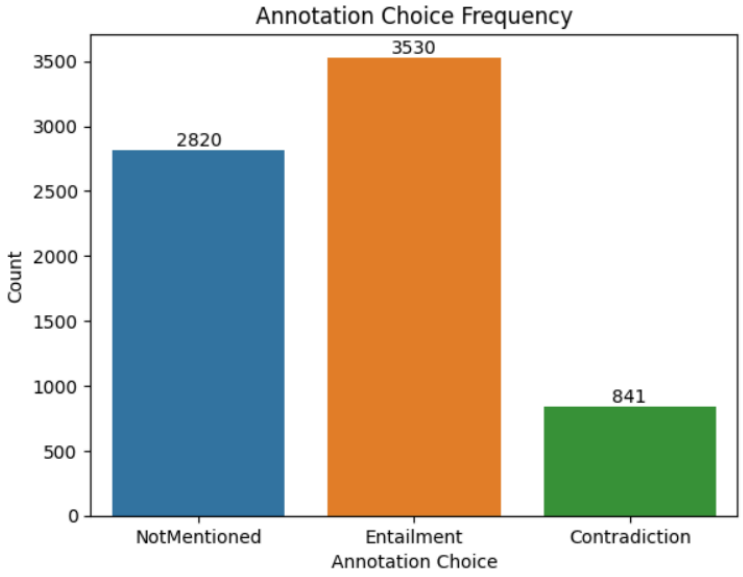
❖ EDA - Exploratory data analysis

➢ There are total 17 hypothesis in train, dev and test split.



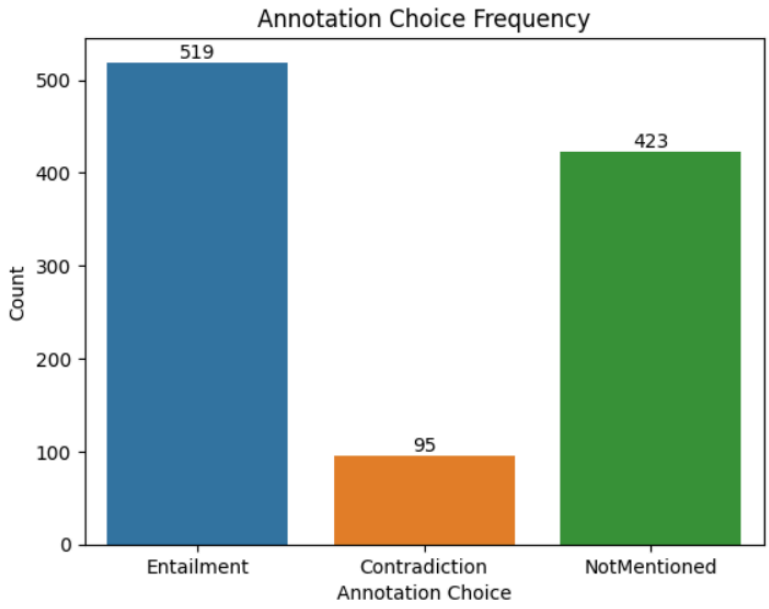
- ➢ There is not a single entry with a NULL value in all three splits so the data is clean.
- ➢ Most of the contract is length 5000 to 15000 (in character). The avg contract length is 11,050 (in character) in train data
- ➢ Also for dev and test data, the average contract length is around 11,200 (in character).
- ➢ Most of the span length is less than 400. Same in dev and test data.
- ➢ The annotation count per contract is 17 for all contracts in train, dev , and test.
- ➢ There are only 3 types of contract:- 'search-pdf', 'sec-text', 'sec-html'.
- ➢ 'Search-pdf' is the most frequent

➢ **<u>Train Dataset Annotation distribution</u>**



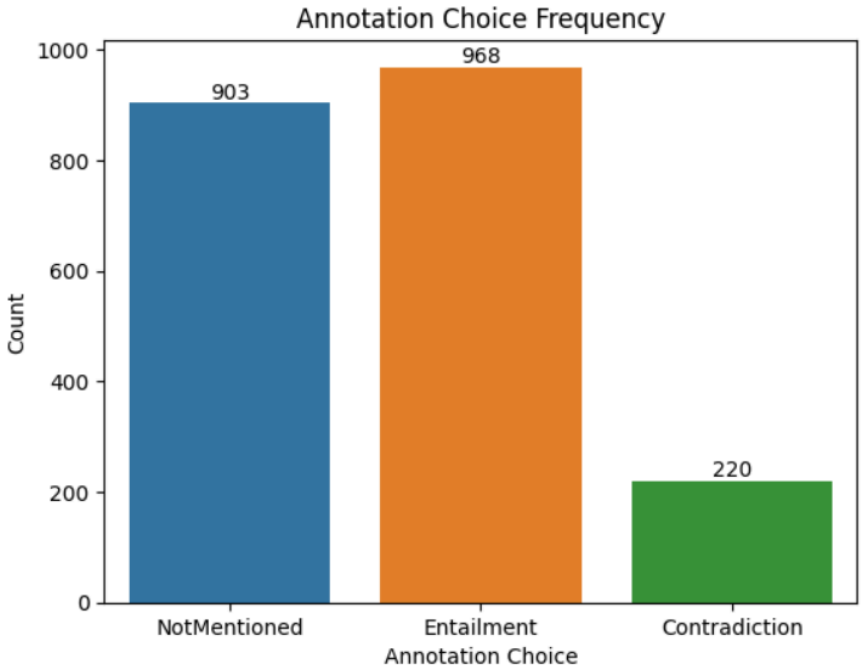
- The train data is imbalanced because the ratio of classes Entailment: Contraction: Not Mentioned is almost 4:1:3.

➢ **Dev Dataset Annotation distribution**



- The dev data is imbalanced because the ratio of classes Entailment: Contraction: Not Mentioned is almost 5:1:4.

➢ **Test Dataset Annotation distribution**



- The test data is imbalanced because the ratio of classes Entailment: Contraction: Not Mentioned is almost 4:1:4.

➢ Loss scaling or undersampling of the oversampled class or oversampling of the undersampled class is required because there is a remarkable class imbalance.

❖ Approach: Exploration of baseline with different LLM

We are investigating the potential of pre-trained large language models (LLMs) to improve performance on the Contract NLI (Natural Language Inference) task, aiming to set a new benchmark. In our initial exploration, we conducted direct inference on test data using models such as GPT-4 and Microsoft's PHI, with the following results:

- **GPT-4:** We tested 67 contracts through API calls and achieved **54% accuracy** on the NLI task. This was a partial test due to reaching our resource limit.
- **Microsoft PHI:** Similarly, we achieved **50% accuracy** using Microsoft's PHI model for the same NLI task.
- **Prompt engineering:** We did prompt engineering to find out the best prompt for the direct inference for both GPT4 and PHI models. We did the manual prompt tunning. we saw a couple of predicted labels then based on how much the model was confused we changed the instruction to reduce ambiguity. So in the future, we will explore more prompt tunning and engineering.

Experiment Details

In this experiment, we utilized simple prompt-based inference without any prior prompt engineering, fine-tuning, or model-specific optimizations. The results suggest potential for significant improvement through the following strategies:

- **LoRA (Low-Rank Adaptation) Fine-Tuning:** LoRA allows efficient fine-tuning of large models by freezing the majority of parameters and introducing low-rank

updates, reducing computational costs while maintaining strong performance. Applying LoRA fine-tuning specific to contract NLI could improve accuracy by leveraging more domain-specific knowledge.

- **Prompt Tuning and Engineering:** Carefully crafted prompts could significantly enhance the performance of these models. By refining prompts to better capture the nuances of contractual language and legal reasoning, the models could make more accurate NLI predictions.

★ Potential Use of LangChain

To enhance the prompt engineering we propose to use Langchain to try different prompt techniques like Chain of Though, Sequential Prompting, Step by Step Contract Analysis etc.

★ <u>Conclusion</u>

Our baseline exploration suggests that while GPT-4 and Microsoft PHI show promising results for contract NLI, there is ample opportunity to improve through fine-tuning and prompt optimization. Integrating LangChain into the workflow could also provide more structured, multi-step reasoning processes, leading to further performance gains. This hybrid approach of fine-tuning, prompt engineering, and structured LLM workflows has the potential to set a new benchmark for Contract NLI tasks.