Syntactic Ambiguity in Legal Language: Automatic Classification and Interpretation

HAMZA WALID AHMED ABDOU ELKADY, University of Twente, The Netherlands

Syntactic ambiguity in legal texts poses a serious risk for misinterpretation, inconsistent enforcement, and legal disputes. This study investigates whether large language models (LLMs) can automatically detect, classify, and interpret syntactic ambiguity in legal sentences. A manually labeled dataset was used to evaluate the classification performance of GPT models and to fine-tune LegalBERT for cost-effective local classification. The best results were with sentences that contained coordination ambiguity as they were the most consistently recognized. For interpretation, Gemini was used to generate paired rewrites of ambiguous sentences, which were then used to fine-tune a T5 model. While the T5 model preserved the intent of most inputs and avoided hallucinations, it often failed to restructure sentences in a way that fully resolved ambiguity. Overall, the study shows promise in using LLMs for ambiguity-related tasks, but highlights that high-quality data and expert guidance are essential for reliably training cost-effective models.

Additional Key Words and Phrases: Syntactic Ambiguity, Legal Texts, Natural Language Processing, Legal Interpretation, Text Generation

1 INTRODUCTION

During drafting of legal documents and policies, precision of language is important to avoid misinterpretation. Tiersma [9] mentions that vagueness and ambiguity occurs in languages and even more in legal language. Syntactic ambiguity occurs when a sentence has multiple interpretations due to its grammatical structure. This ambiguity can lead to different interpretations of legal obligations and rights with potentially serious legal, ethical, and financial consequences. Courts and regulatory agencies depend on textual clarity to apply the law consistently. Misinterpretation due to ambiguity can lead to inconsistent enforcement and expose organizations to unintended liability. For instance:

- *This Regulation should be without prejudice to the possibility for Member States to lay down the requirements as to the powers granted to the supervisory authority* ***to bring infringements of this Regulation to the attention of judicial authorities and to engage in legal proceedings, in particular when it is necessary to safeguard data protection rights****.*

Does the following phrase "in particular when it is necessary to safeguard data protection rights" modify the phrase "to engage in legal proceedings" or the whole compound action *(in bold)*. The different possibility of modifications leads to different interpretations and consequences. If the only phrase modified is "to engage in legal proceedings", then the supervisory authority can always inform judicial authorities, but can not always initiate legal proceedings. If both clauses are modified, the authority is only allowed to inform judicial bodies and take legal action when it is necessary to safeguard data protection rights.

Previous research shows that legal experts' judgments about ambiguity are often subjective and often influenced by their ideological stances [4]. When legal experts were asked whether ordinary readers would agree on the meaning of a statute, they were much more likely to find it ambiguous from the perspective of an ordinary reader.[4] .

Judges often misunderstand linguistic principles [7] ; as a result, decisions made by the judges are inconsistent and may lead to unfairness in legal outcomes.

Ambiguity can also have a great impact on contract drafting. Adams [1] warned that many common drafting habits include vague connectives and modifiers like "and/or, every, each, and any" or plural nouns can cause ambiguity. This ambiguity can then lead to costly disputes.

1.1 Problem Definition

These findings highlight that not only ordinary people, but judgments by legal professionals are also often distorted when encountering ambiguous texts. Syntactic ambiguity leads to biases in interpretation, resulting in inconsistencies in legal decision-making and enforcement. This creates a critical need for computational systems that can automatically detect and classify syntactic ambiguities in legal documents provide further clarity and fairness.

2 RESEARCH QUESTION

To address the challenges discussed in the preceding section, this project is aims to answer the following research question:

"To what extent can a system automatically detect and classify syntactic ambiguity in legal texts, and generate accurate alternative interpretations that align with human reasoning in a cost-effective manner?"

This research questions contains three main goals:

- Detecting and classifying syntactic ambiguities
- Generating alternative interpretations for ambiguous legal sentences.
- Exploring scalable and cost-effective methods for achieving these tasks

3 RELATED WORK

This section reviews prior work on computational efforts to detect and resolve ambiguities using machine learning and language models. Understanding syntactic ambiguity in legal texts requires insights from legal interpretation and linguistics. Additionally, this section goes over models that can be used to achieve the goals mentioned in the Research Question section.

TScIT 43, July 4, 2025, Enschede, The Netherlands

A study aimed to investigate BERT's [2], the transformer model, performance on legal tasks. As a result, LegalBERT [2] was introduced, a domain-adapted transformer model which is pretrained on English legal texts such as contracts, court cases, and legislation. Although LegalBERT achieved improved performance on legal document classification, it was not designed specifically to detect syntactic ambiguity.

Shifting from NLP models focused on detection and classification, there are models that can help approach on generating interpretations. T5 [6], introduced a unified text-to-text framework capable of handling generation tasks, including paraphrase generation. Due to T5 being pre-trained on C4 dataset (hundreds of gigabytes of English text from the Web), fine-tuning is required in order to generate alternative interpretations of ambiguous legal sentences correctly.

The concept of "law smells", are problematic patterns in legal drafting that may signal ambiguity, complexity, or poor structure [3]. One of these "smells", ambiguous syntax, is defined as the use of logical or control flow operators (e.g., "and", "or", "and/or"), control flow operators (e.g., if, else, and while), or punctuation (e.g., commas and semicolons) in such a way that allows for multiple interpretations. A set of regular expressions (Regex) was designed to identify common ambiguous constructions. However, the study notes the method used is over-inclusive and that some legal experts should review whether flagged sentences are truly ambiguous [3]. This validates the practical need for more advanced methods of classification and interpretation. However, the provided datasets cannot be used to evaluate LLM performance due to it being over-inclusive.

A benchmark called AmbiBench [8] was introduced to investigate how both humans and language models interpret task ambiguity in classification tasks. Findings show that model scaling alone is insufficient. Moreover, models require both scale and human feedback training to perform comparably to humans. Although the work focuses on task ambiguity rather than syntactic ambiguity in natural language, their results help shape this project's methodology.

Prior work demonstrates that language models can benefit significantly from fine-tuning on domain-specific ambiguous examples, and that human feedback is a vital factor in improving the model's performance.

4 METHODOLOGY

This research addressed three core challenges: (1) detecting and classifying syntactic ambiguity in legal texts, (2) generating accurate alternative interpretations of ambiguous sentences, and (3) exploring cost-effective methods to accomplish both of the previous two challenges. For this, the methodology combined evaluating commercial large language models (LLMs), such as GPT, on classifying sentences with ambiguity and generating multiple interpretations for ambigious sentences. The LLMs were then used to assist the fine-tuning of open-source models, such as LegalBERT. Fine-tuning open-source models lowers the cost of the following tasks: detecting, classifying, and generating interpretations. Moreover, it allows organizations to process sensitive legal texts without sending data to external APIs. In addition, it allows the models to become tailored to domain-specific needs.

4.1 Ambiguity Detection and Classification

The first objective was focused on identifying whether there is ambiguity in a given sentence, and if so, classifying the type of ambiguity. The types of ambiguity focused on in this study are **Attachment ambiguity** and **Coordination ambiguity**. An initial dataset was built from legal sources. The dataset was then manually classified as one of the following: "Attachment ambiguity", "Coordination ambiguity", or "Non-ambiguous" *(see Appendix A for defintions and example sentences of each category)*. The manually labeled entries were then used to evaluate GPT's API labeling performance as the ground truth. GPT was then prompted to label the ambiguity types across the dataset. These results were compared against the manual labels. LegalBERT was then fine-tuned using a larger GPT-labeled dataset to evaluate whether it could serve as a local and cost-effective alternative to LLMs.

4.2 Alternative Interpretation Generation

Gemini was used to generate example interpretations using sentences from the original dataset. The generated interpretations were manually reviewed on their quality and structure. Based on the quality of these interpretations, a lightweight generative model (T5) was fine-tuned using a dataset containing ambiguous sentences paired with multiple Gemini generated and plausible interpretations.

5 AMBIGUITY DETECTION AND CLASSIFICATION

5.1 Manual Dataset Construction

An initial dataset of legal sentences was created by manually extracting text from the General Data Protection Regulation (GDPR). The goal was to construct a dataset showing a balanced range of examples of different types of syntactic ambiguity common in legal writing. Each sentence was manually labeled as either attachment ambiguity, coordination ambiguity, or non-ambiguous.

Labeling was carried out over three rounds. In the first round, annotations were made based on surface cues, contextual interpretation, and the working definitions of each category. Upon reviewing the results, two issues emerged: some labels were applied inconsistently across similar sentence types, and there was an imbalance across the categories.

To address this, the second round introduced a more structured labeling rubric with clearer decision criteria. Using these guidelines, the dataset was re-evaluated to improve consistency across annotations.

In the third round, the focus shifted to achieving a more balanced distribution. Additional examples were added, in particular for attachment ambiguity, which had not been found in the same quantities as the other categories. Duplicate sentences were removed and replaced. The resulting dataset offered a more even representation of all three categories, making it better suited for reliable evaluation.

The final dataset included 100 entries, distributed as follows:

- Attachment ambiguity: 33 sentences
- Coordination ambiguity: 29 sentences
- Non-ambiguous: 38 sentences

This balance was necessary to ensure that subsequent evaluation of the OpenAI API's labeling capabilities would be tested equally across the three categories.

5.2 GPT Labeling procedure

The GPT labeling procedure went through an iterative process as well. The same dataset was initially labeled using Open AI's **GPT-3.5 Turbo (Knowledge cut off: Sep 01, 2021)** via API. The prompt provided only contained definitions of each category (*see Appendix B.1*). The GPT annotated dataset was then evaluated based on the manual annotated dataset. Each sentence was submitted individually to the model, rather than the whole dataset at once, to ensure that GPT evaluated each input in isolation. As a result, bias is reduced due to the decrease of influence by earlier sentences. However, the F1 weighted avergae score was 0.46 which was too low if it was used to annotate a larger dataset. The entries that were incorrectly annotated were all manually analyzed. Moreover, common error patterns that lead to misclassifications and how to avoid them were noted. The misclassifications revealed a substantial number of sentences that were ambiguous due to keywords (e.g., "which" or "such") or complex phrases that contained many "and"'s even when the sentence was clear. This lead to only 15 out of 38 of the non-ambiguous sentences being labeled correctly. The dataset was then labeled once again, but this time using Open AI's **GPT-4o (Knowledge cut off: Oct 01, 2023)** and with a refined prompt. The refined prompt (*see Appendix B.2*) included examples and clarifications on how to deal specific sentence structures to help aid GPT in the annotation process alongside the definitions of the categories. An F1 weighted average score of 0.73 and an accuracy of 73% was then achieved [1] . The classification performance for each category can be found in Table 1. In addition, the accuracy of detection of ambiguity was 92%.

Table 1. True Positives by Category Type

Ambiguity Type	**Correct**	**Total**
Non-ambiguous	26	38
Attachment ambiguity	23	33
Coordination ambiguity	24	29

The F1 score and accuracies were deemed high enough to move to the next step of the phase which is scaling GPT's labeling over a larger dataset.

5.3 Dataset Scaling

The first step of scaling the labeling process is to build a larger dataset while ensuring that there are enough entries of each category. Ensuring there is a sufficient amount of each entry ensures the LegalBERT model has enough examples to learn the **structural patterns** and **syntactic reasoning** of each type of category in order to correctly classify it.

[1]As part of the prompt refinement process, an additional experiment was conducted to test the use of confidence scores. The confidence scores were based on on how strongly a sentence matched a rubric for ambiguity. The idea was to not give the presence of surface-level cues significance in the grading. However, this did not impact the F1 score as it was approximately 0.73 as well

To accelerate the process of building the dataset an automated scraping pipeline was implemented to extract legal text from EU regulations. HTML content was downloaded from regulation pages and processed using Python's *requests*, *BeautifulSoup*, and *re* libraries and saved to a CSV file. Raw sentences were then further preprocessed manually to remove lists and numbered bullet points (e.g., "1.", "a)", etc.), which could negatively impact BERT's sentence embeddings. This was done due to transformer models relying on continuous, well-formed sentence inputs.

The large dataset was then ran through the same prompt, as the smaller dataset, by GPT-4o. Each individual entry was again submitted to GPT-4o rather than the full dataset all at once and the annotated data set was then checked to make sure there were enough examples of each category for the fine-tuning step.

5.4 Fine-Tuning LegalBERT

LegalBERT-Base (legal-bert-base-uncased) was then fine-tuned with this large dataset using the HuggingFace **Trainer** API built on top of PyTorch. The dataset used for fine-tuning consisted approximately 1,000 sentences labeled using GPT annotations. To ensure fair evaluation the dataset was divided into three subsets: 80% for training, 10% for validation, and 10% for testing. This was done to ensure that the model has enough data to learn effectively (training set), while reserving separate subsets for tuning hyperparameters (validation set) and measuring generalization performance (test set).

To further the reliability of the evaluation, the data was split using the "**stratified sampling**" method. In stratified sampling, the data is divided such that the class distribution remains proportional across all subsets. Without stratification, random sampling might lead to class imbalance in the validation or test sets (e.g., too few coordination examples in the test set), which would distort performance metrics. As a result, the evaluation becomes more reliable and reducing bias introduced by class imbalance.

The best model was selected based on lowest validation loss, even though later epochs showed marginal improvements in accuracy. This strategy prioritizes generalization performance over raw training fit.

5.5 Evaluation Metrics

To evaluate the model's performance on the ambiguity classification task, the standard classification metrics including accuracy, precision, recall, and macro-averaged F1 score were computed . The best performing model, selected based on validation loss, had 82% accuracy. Although later epochs showed marginal improvements in accuracy, selecting the model based on validation loss prioritizes generalization performance over raw training fit.

Table 2 shows the precision, recall, and F1 scores per class. The model performed best on non-ambiguous sentences and coordination ambiguity, but was slightly weaker for attachment ambiguity based on the F1 scores. Attachment ambiguity is the only category to have *Precision* lower than *Recall*. This suggests the model tends to over-predict attachment ambiguity likely flagging syntactic structures like prepositional phrases or modifiers even in cases that are actually unambiguous. Although the refinement of the GPT prompt significantly reduced the over-prediction of ambiguity, the improvement was not uniform across both attachment and coordination ambiguity on a larger scale.

While the issue of over-predicting ambiguity was significantly reduced during the GPT labeling phase in *in Section 5.2*, this improvement was not uniform across both attachment and coordination ambiguity on a larger scale. The most notable and consistent gains were observed in coordination ambiguity.

Table 2. Classification report: precision, recall, and F1-score per class.

Class	Precision	Recall	F1-Score	Support
Non-ambiguous (0)	0.93	0.75	0.83	36
Coordination ambiguity (1)	0.80	0.80	0.80	30
Attachment ambiguity (2)	0.69	0.80	0.74	41
Accuracy		0.79		107
Macro avg	0.81	0.78	0.79	107
Weighted avg	0.80	0.79	0.79	107

The confusion matrix in Figure 1 further supports this: misclassifications are most common between the non-ambiguous and attachment categories. An important observation highlighted in the confusion matrix is that although some sentences with coordination ambiguity were classified as attachment ambiguity, none of them were classified as non-ambiguous. This is significant because, when the model receives sentences with coordination ambiguity, it recognizes the presence of ambiguity. In addition, Figure 1 shows that the model has a 97% accuracy in ambiguity detection, correctly identifying 69 out of the 71 truly ambiguous sentences. While this percentage seems to be strong, it is inflated due to the model over-labeling sentences as ambiguous.

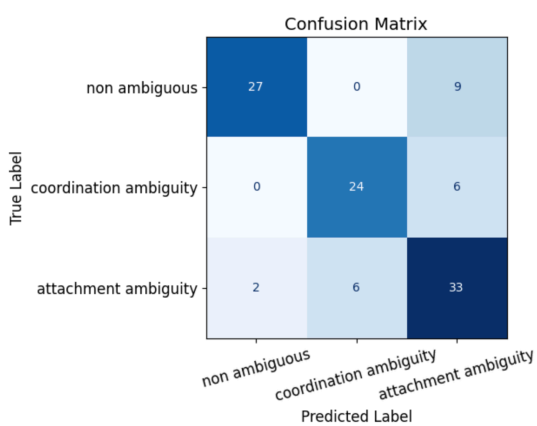

Fig. 1. Confusion matrix showing classification performance across categories on the testing subset.

5.6 Error Analysis and Behavioral Testing

While the overall classification metrics, in particular accuracy and macro F1, suggest strong performance, the confusion matrix highlights that 25% of the Non-ambiguous predictions by the model were truly Attachment ambiguity. In addition, the F1 score of the Attachment ambiguity is not as high as the other categories.

To better understand the root causes of these errors, a series of behavioral tests were designed. The tests included new and different sentence structures created by ChatGPT. All the sentences were manually reviewed and edited to ensure they are sentences with legal context and are correctly labeled. The sentences were either labeled as "Non-ambiguous" or "Attachment ambiguity" in order to address the errors from the evaluation metrics.

The first test was done to address the problem of the Attachment ambiguity's F1 score. The same set of structurally ambiguous sentences was presented to the model in two forms: one with commas inserted around the modifying phrase, and one without. These controlled pairs were designed to isolate the impact of punctuation without changing the sentence's underlying structure or meaning. The sentences were also designed to potentially confuse the model into predicting Non-ambiguous or Coordination ambiguity. An example of a sentence written in 2 ways used:

- **Without commas:** All data exports must be logged by system administrators on Tuesdays.
- **With commas:** All data exports must be logged, by system administrators, on Tuesdays.

There was a 30% increase correct attachment ambiguity classification. This suggests that the model is highly sensitive to punctuation for syntactic disambiguation. In addition, the model may not be learning deep syntactic relationships, but rather responding to surface-level formatting cues.

The second test was done to understand when are non-ambiguous sentences classified as sentences with attachment ambiguity. This test consisted of 100 examples of non-ambiguous sentences distributed evenly across four different syntactic forms:

- Prepositional Phrase
- Time-Based Clauses
- Phrases with 'With'
- Legally Precise but Syntactically Dense

The examples of the four forms were chosen to be structurally clear, despite them occurring in ambiguous constructions (*an example of each form can be found in Appendix C*). The accuracy of basic prepositional phrases was 100%. On the other hand, the model misclassified 80% of phrases with "with" and 40% of the time-based clauses and syntactically dense examples as ambiguous. This pattern suggests that the model is not failing on a single syntactic structure, but is instead over-sensitive to surface-level patterns it learned from ambiguous training data. Similarly to the previous test, this test shows these errors are likely the result of overgeneralization, where the model relies on pattern recognition rather than true syntactic reasoning. The dataset does not contain enough sentences that contain patterns found in ambiguous sentences but are non-ambiguous.

6 ALTERNATIVE INTERPRETATION GENERATION

6.1 Fesibility of Gemini Interpretation Generation

Due to cost constraints over the long run with text generation using the OpenAI API, Gemini's model was selected for this phase. 50 ambiguous entries from the manually labeled dataset (from the classification phase) were used to evaluate the quality of Gemini's interpretations. There were 25 sentences of each type of ambiguity. The model was not only tested for the quality of the interpretations but also for the reasoning behind the distinct interpretations. Each sentence was submitted individually to the Gemini API along, and interpretations and reasoning were evaluated based on human judgment. Along side each sentence the type of ambiguity was also passed as an input to evaluate it's effect on the output. The prompt used by Gemini's model can be found in Appendix D. An example of the output can be found in the Figure 2.

Sentence:
The controller shall provide a copy of the personal data undergoing processing.
Ambiguity Type: Attachment Ambiguity

Explanation:
The ambiguity in the sentence lies in the attachment of the phrase *"undergoing processing."* It is unclear whether this phrase modifies *"personal data"* or *"a copy."*

Interpretation 1:
The controller shall provide a copy of the personal data that is undergoing processing.
This interpretation attaches *"undergoing processing"* to *"personal data."* It suggests the controller only needs to provide a copy of the data currently being processed. The focus is on the **status of the data**.

Interpretation 2:
The controller shall provide a copy of the personal data which is undergoing a processing operation (e.g., a copy that is itself being processed or copied).
Here, the phrase modifies *"a copy,"* implying the copy itself must be undergoing some processing. This is a less likely, but grammatically valid, reading—particularly in technical contexts involving data handling protocols.

Fig. 2. Example of Gemini's generated interpretation for a sentence with attachment ambiguity.

After analyzing all the outputs it was clear that Gemini's interpretation correctly identified the source of ambiguity. However, the underlying ambiguity is not clearly resolved. The changes must be accompanied by explanations to make interpretations become more distinct. As shown in the example of the output in Figure 2, rewritten sentences made slight changes such as adding "that" or "which" to make the possible distinction. Since many examples did show contrastive structure, the next step would be aimed to test whether a generation model could begin learning from even partially contrastive data.

6.2 Dataset Construction

The large dataset constructed in *Section 5.3* was passed through Gemini's model, however all non-ambiguous entries were replaced with new ambiguous entries from more legal documents. The prompt was then modified slightly to ensure all sentences are written correctly and the interpretations are sorted into two separate columns with no further explanations. Each separate sentence was again passed one at a time to Gemini. After completion the final data set was also checked for possible entries with the same two interpretations to prevent generation of duplicate interpretations. However, the model did not generate any duplicate interpretations.

6.3 Fine-Tuning T5

For the interpretation generation task, a T5-small model from *HuggingFace* was fine-tuned in a standard sequence-to-sequence setup. The dataset was first flattened. In other words, instead of using a single output per sentence, the data was doubled by using the two separate rewrites. As a result, the model learns that one sentence can have multiple plausible interpretations. T5 is trained to generate a single target sequence for each input so, having two logically distinct outputs would cause the model to memorize the formats in the dataset. Flattening eliminates issues such as interpretations being valid, but the order being flipped is considered wrong. Evaluation was performed using ROUGE-L [5] and after training it restores the model checkpoint with the best ROUGE-L score.

6.4 Metric Evaluation

While ROUGE [5] is commonly used in text generation tasks such as summarization, it has limitations in the context of interpretation generation. Multiple outputs may be considered valid and disambiguation depends on the grammar rather than text overlap. However ROUGE-L scores are still important as it is sensitive to long-sequence overlap. Moreover, ROUGE-L score indicates if the generated output is within the same context of the reference interpretation. The score achieved was 0.51. This indicates the generation preserves relevant information from the source but may not match reference interpretations word-for-word.

Regarding the training loss and validation loss, the gap between them is small (0.03) which indicates there is no significant overfitting. However, this does not guarantee good performance on unseen data. As a result, manual tests must be carried to conclude the model's performance and how meaningful the generated interpretations are.

6.5 Manual Evaluation

The test set consisted of 32, ChatGPT generated and classified, sentences evenly divided into four categories: coordination ambiguity, attachment ambiguity, non-ambiguous, and edge cases. Every generated sentences were manually reviewed to ensure they accurately represent each category and that they are correctly classified. The sentences with coordination and attachment ambiguity were simple sentences that tested basic functionality using classic syntactic structures that trigger ambiguity such as use of 'and' and 'or' conjunctions or phrases with prepositions like 'with' or "by'. Non-ambiguous sentences were used to test the model's behavior compared to Gemini when it encounters sentences with no ambiguity. Does it rewrite the sentence in two entirely different ways unlike Gemini? Does it not produce any interpretation like Gemini ? Does it just produce minor formatting/paraphrase differences? Edge cases were used to test the model's performance when encountering complex clause structures that contain ambiguity. Complex structures include sentences with multiple instances of "with" clauses or the use of coordination modifiers along with clauses with attachment ambiguity.

The results on the coordination ambiguity set showed the model was aware that punctuation (commas) can shift grouping scope, but it never learned to explicitly restructure coordination. This means it did not learn to rephrase ambiguous coordinated structures into unambiguous forms with clearly separated grouped parts. The example below demonstrates how the T5 model uses commas for grouping scope:

The company may collect, store and share data or destroy it upon request.

- The company may collect, store, and share data or destroy it upon request.
- The company may collect, store, and share data, or destroy it upon request.

The example below shows how rephrasing the sentence can result in a more clear interpretation:

- The company may either collect, store, and share data, or delete it upon request.

In sentences with attachment ambiguity, the model was not able to move the prepositional phrase from one head to another. This indicates that the model lacked syntactic awareness.

In non-ambiguous sentences the model did not hallucinate ambiguity and mostly maintained the structure of the original sentence. This indicates the model knows when not to over generate. The example below shows the output of the model when encountering non-ambiguous sentences:

The report must be filed within thirty days.

- The report must be filed within thirty days.
- The report should be filed within thirty days.

Although 'must' and 'should' are often treated as similar in everyday language, they carry distinct implications. "Must" implies obligation, while "Should" implies a recommendation.

The model struggled with edge cases, often preserving the general idea or intent of a sentence. However, these subtle shifts in wording can have a large impact in high-stake cases where strength of the obligation must be clear.

There were two additional issues noticed while evaluating the model. First, as made evident through the examples above, the first interpretation is identical in almost all cases to the original sentence. The second issue was that regardless of the type of ambiguity given that is given as an input, the model won't change the interpretations given. However, the model did well across all tests by not inventing completely new meanings or dropping clauses. The model was also able to use commas as shallow disambiguators in sentences with clear coordination ambiguity. On the other hand, there no structural correction in all attachment and most coordination cases. In addition, edge cases went mostly untouched.

7 DISCUSSION AND FUTURE WORK

7.1 Effectiveness of Automated Ambiguity Resolution

Initially GPT's output was heavily influenced by surface-level indicators when labeling attachment ambiguity and coordination ambiguity. After the refinement of the given prompt only cases of coordination ambiguity were sufficiently classified on a large scale. GPT focused on finding grouping issues and not just classify any sentence with "and" or "or" as coordination ambiguity. This improved performance aligns with the 'and...or' syntactic structures flagged by Coupette et al. [3]. While their method does not annotate confirmed ambiguity, it suggests LLMs can detect coordination ambiguity based on context and that they are more precise as they do not just depend on surface-level indicators for this specific ambiguity.

In contrast, attachment ambiguity proved more difficult for the model to classify accurately. There is a strong chance that sentences which have clauses that start with words like "with" contain attachment ambiguity. However, attachment ambiguity is not a guarantee as there also has to be a plausible alternative structure. The reliance on shallow cues became noticeable during large-scale labeling, which carried over into the training data used to fine-tune LegalBERT. As a result, the percentage of non-ambiguous classified as attachment ambiguity increased slightly in the final model.

Some of Gemini's interpretations showed minor semantic drift (*as shown in Figure 2*), where the model altered the meaning slightly to produce a second variant. These shifts were subtle but could be legally significant, particularly in high-stakes contexts. This would later on make it more difficult for the the T5 model to learn the patterns or small differences that cause clearly distinct interpretations. There is no standardized evaluation metric to assess the accuracy of the interpretations. Since interpretations can be written in a variety of different ways it becomes difficult to benchmark model performance in a meaningful way.

A limitation that had a significant impact on both phases (classification and generation) was the difficulty of collecting syntactically ambiguous legal sentences at scale. There are no already existing datasets that specifically target the identification and resolution of syntactic ambiguity within legal language. As a result, making dataset creation a largely manual and time-consuming process. Moreover, datasets that are either too small to support generalization, or too noisy when scaled without extensive human review. Lack of professional review over the annotated dataset, may result in the model to overfit to surface features that appear ambiguous to general readers but are legally unambiguous.

In the classification phase, the restricted diversity of examples caused the LegalBERT model to focus on sentences with explicit relative clauses rather than learning to generalize broader syntactic patterns. In the generation phase, the lack of parallel examples showing contrasting interpretations resulted in the model paraphrasing. Paraphrasing only shows the model's fluency, but has no ability for structural reasoning. In the generation phase, the dataset was not large enough for the model to identify the patterns to generate two distinct interpretations.

7.2 Feasibility and Future Directions

Many aspects of the methodology could be improved to strengthen the results. First, incorporating some form of human spot-checking or disagreement filtering could have helped mitigate the issue of over-relying on surface features during large-scale annotation. Second, although it is important that general readers understand all sentences in legal texts, incorporating expert validation during the construction of the datasets would have helped. This ensures that annotated ambiguity cases were not only grammatically plausible but also legally meaningful. Legal professionals could have reviewed a portion of the dataset to identify over-labeling or misinterpretations that might not reflect actual risk in practice. Third, a larger and more diverse dataset would benefits both T5 and LegalBERT models by allowing them to identify a broader range of linguistic and legal variations.

LegalBERT, in particular, has shown strong potential in identifying and labeling ambiguous sentences. The current performance in flagging potentially ambiguous legal texts is promising. LegalBERT could become a valuable tool in pre-screening legal documents for interpretive risk.

8 CONCLUSION

The overall results show that there is potential for LLMs to classify and interpret ambiguity. However, keeping the process automated in a more cost-effective way is more difficult. Identifying potentially ambiguous sentences can be significantly improved with minor changes in the methodology. On the other hand, classifying the sentences by ambiguity and generating interpretations requires even further improvements. Moreover, the study also revealed the difficulty that LLMs face when creating large datasets, for the interpretation generation task, to help fine-tune cost-effective models. The datasets they create are not structured nor expressive enough to teach models meaningful syntactic distinctions. This is in addition to training models on identifying, classifying and interpreting ambiguity, requiring the supervision of legal experts, large datasets, and careful construction of those datasets. These findings point to the need for hybrid approaches that combine LLM capabilities with targeted linguistic guidance.

REFERENCES

[1] K.A. Adams. 2004. *A Manual of Style for Contract Drafting*. ABA, Section of Business Law. https://books.google.nl/books?id=n7-jxMskU9MC

[2] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559* (2020).

[3] C. Coupette, D. Hartung, J. Beckedorf, et al. 2023. Law Smells. *Artificial Intelligence and Law* 31 (2023), 335–368. https://doi.org/10.1007/s10506-022-09315-w

[4] Ward Farnsworth, Dustin F. Guzior, and Anup Malani. 2010. Ambiguity about Ambiguity: An Empirical Inquiry into Legal Interpretation. *Journal of Legal Analysis* 2, 1 (03 2010), 257–300. https://doi.org/10.1093/jla/2.1.257 arXiv:https://academic.oup.com/jla/article-pdf/2/1/257/23562589/2-1-257.pdf

[5] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[6] 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 (2020), 1–67. http://jmlr.org/papers/v21/20-074.html

[7] Lawrence M. Solan. 1993. *The Language of Judges*. University of Chicago Press.

[8] Alex Tamkin, Kunal Handa, Avash Shrestha, and Noah Goodman. 2022. Task Ambiguity in Humans and Language Models. arXiv:2212.10711 [cs.CL] https://arxiv.org/abs/2212.10711

[9] Peter M Tiersma. 1999. *Legal language*. University of Chicago Press.

A CATEGORY DEFINTIONS AND EXAMPLES

A.1 Attachment Ambiguity

- *Definition:* Occurs when a phrase can attach to more than one part of a sentence, changing its meaning.
- *Example:* “Any part of such a declaration which constitutes an infringement of this Regulation shall not be binding.”
  Explanation: is the part of the declaration that is infringing only non-binding, or is the entire declaration non-binding when a part of it is infringing

A.2 Coordination Ambiguity

- *Definition:* Occurs when it’s unclear how conjunctions like “and” or “or” group elements, leading to multiple valid interpretations.
- *Example:* “Processing is necessary for the purposes of preventive or occupational medicine, for the assessment of the working capacity of the employee, medical diagnosis, the provision of health or social care or treatment or the management of health or social care systems and services on the basis of Union or Member State law or pursuant to contract with a health professional and subject to the conditions and safeguards referred to in paragraph 3.”
  Explanation: Does the phrase “on the basis of Union or Member State law or pursuant to contract...” apply only to the final clause (“management...”) or to all listed purposes?

A.3 Non-Ambiguous

- *Definition:* A sentence with a single and clear interpretation.
- *Example:* "Paragraph 1 shall not apply if the decision is based on the data subject’s explicit consent."
  Explanation: No structural confusion

B GPT PROMPT EXAMPLES

B.1 Initial Prompt (GPT-3.5)

You are a legal linguistics expert. Use the following definitions and examples to classify each sentence.

Definitions:

- **Attachment ambiguity:** A clause or prepositional phrase could attach to more than one part of the sentence, and the meaning would change depending on what it attaches to.
- **Coordination ambiguity:** Lists joined with “and,” “or,” or “and/or” create unclear grouping or scope.
- **Non-ambiguous:** The sentence is structurally and legally clear and does not meet either of the above conditions.

B.2 Refined Prompt (GPT-4o)

You are a legal linguistics expert. Use the following definitions and examples to classify each sentence.

Definitions:

- **Attachment ambiguity:** A clause or prepositional phrase could attach to more than one part of the sentence, and the meaning would change depending on what it attaches to.

- **Coordination ambiguity:** Lists joined with "and," "or," or "and/or" create unclear grouping or scope.
- **Non-ambiguous:** The sentence is structurally and legally clear and does not meet either of the above conditions.

Clarifications:

- Label a sentence as attachment ambiguity if a clause like "which..." could grammatically modify more than one part of the sentence and that difference would change the legal interpretation.
- Do not label a sentence as ambiguous simply because it includes relative clauses (e.g., "which...") unless more than one attachment is grammatically plausible and meaningfully distinct.
- Do not label coordination ambiguity if the coordinated elements are clearly parallel and unambiguous in meaning or scope.
- If the sentence is a legal fragment without a main verb, do not label it ambiguous unless it creates real syntactic confusion.

Examples:

(1) *Sentence:* "The controller shall provide a copy of the personal data undergoing processing."
Label: Attachment ambiguity
(2) *Sentence:* "The contact details of the data protection officer, where applicable."
Label: Non-ambiguous
(3) *Sentence:* "The data subject shall have the right to request access and rectification or erasure of personal data."
Label: Coordination ambiguity
(4) *Sentence:* "Any provision of the agreement which violates applicable law shall be void."
Label: Attachment ambiguity

Instruction: Only return one of the following labels exactly: **Attachment ambiguity**, **Coordination ambiguity**, **Non-ambiguous**.

C NON-AMBIGUOUS SENTENCE GROUPS EXAMPLES

C.1 Prepositional Phrase

- *Sentence:* "The regulator approved the policy to the supervisory authority.",

C.2 Time-Based Clauses

- *Sentence:* "The organization updated the policy after consulting stakeholders.",

C.3 Phrases with 'With'

- *Sentence:* "The user accessed the portal in compliance with regulations.",

C.4 Complex but Structurally Clear

- *Sentence:* "The agreement was executed with mutual consent and in accordance with governing law.",

D GEMINI PROMPT

You are a legal linguistics expert. The following sentence contains **(ambiguity-type)**. Generate two distinct and plausible legal interpretations based on the ambiguity-type. Make sure to highlight where the ambiguity occurs and why it causes the 2 interpretations. The 2 interpretations should be 2 ways of writing the given sentence. Sentence: **"sentence"**

Format:

- Interpretation 1: ...
- Interpretation 2: ...

E USAGE OF AI

This report benefited from the use of AI tools, ChatGPT, to improve the clarity and precision of the writing, enhance the flow of ideas, and assist in planning the overall structure. The content acquired from ChatGPT were then reviewed and edited to fit in with the rest of the paper.