Extracting a Prototypical Argumentative Pattern in Financial Q&As

Anonymous ACL submission

Abstract

Argumentative patterns are recurrent strategies adopted to pursue a definite communicative goal in a discussion. For instance, in Q&A exchanges during financial conference calls, a pattern called Request of Confirmation of Inference (ROCOI) helps streamline conversations by requesting explicit verification of inferences drawn from a statement. Our work presents two ROCOI extraction approaches from interrogative units: sequence labeling and text-to-text generation. We experiment with multiple models for each task formulation to explore which models can effectively and robustly perform pattern extraction. Results indicate that machine-based ROCOI extraction is an achievable task, though variation among metrics that are designed for different evaluation axes makes obtaining a clear picture difficult. We find that overall, ROCOI extraction is performed best via sequence labeling (Token-level $F_1 = 0.31$), though with ample room for improvement. We encourage future work to extend the study to new argumentative patterns.

1 Introduction

An argumentative pattern is a recurrent and identifiable structure with a specific function in an argumentative discussion. Such a pattern offers valuable insights into the reasoning processes and dialectical strategies employed by interlocutors in argumentative discourse.

Extracting argumentative patterns from natural discourse presents a significant challenge in the field of Argument mining (AM) (Lawrence and Reed, 2019). Typically, AM involves three stages: (1) the identification, segmentation, and classification of argumentative discourse units (ADUs) (Ghosh et al., 2014), (2) the characterization of the relations between ADUs (Peldszus and Stede, 2013), and (3) the identification of argument schemes, which denote implicit and explicit inferential relations within and across ADUs (Macagno and Walton, 2014). This area of research is often challenged by the idiosyncrasies of spoken language. For instance, in Earnings Conference Call (ECC) Q&A sessions, argumentative content is often embedded in complex statements aimed at maximizing information content while minimizing exchanges (Keith and Stent, 2019). Instead of employing a typical end-to-end AM pipeline, leveraging linguistic patterns that are clearly identifiable as part of argument schemes could be useful for locating argumentative moves, unraveling the complexities in such dialogues.

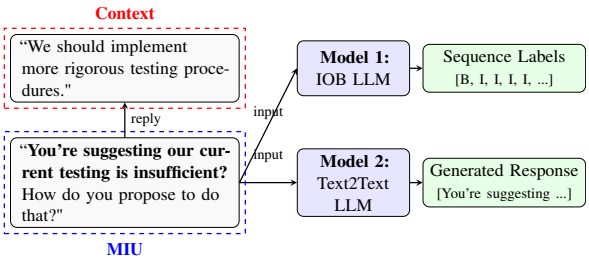


Figure 1: Example ROCOI and the two extraction approaches.

In this paper, we present a novel task and approach to the extraction of a prototypical argumentative pattern called the **Request Of Confirmation Of Inference** (ROCOI). Our work focuses on this argumentative pattern that emerges in questions and presents an easily identifiable surface structure that complements the underlying argumentative function. By deliberately integrating linguistic knowledge into the extraction process, we move beyond the analysis of entire discourse units, instead allowing us to localize ROCOIs inside dialogues. This approach allows us to maintain precise control over pattern detection while dealing with the inherent complexity of argumentative texts.

We specifically focus on the ROCOI pattern as it represents an ideal proto-pattern for exploring how well NLP methods can extract them from

text. These patterns exhibit clear characteristics that make them readily identifiable by trained human annotators, including their interrogative nature, explicit marking of prior reasoning, and requests for confirmation of inferential conclusions. This clarity provides an excellent starting point for developing and evaluating automated extraction methods.

Our experiments encompass two task formulations, comparing an extractive (sequence labeling) and abstractive (text-to-text generation) paradigm. Comparing these two approaches allows us to bridge traditional boundary-marking techniques (Eger et al., 2017; Kuribayashi et al., 2019; Bao et al., 2021) with state-of-the-art language modeling approaches (Raffel et al., 2020; Gorur et al., 2024).

This work represents a crucial step toward the broader goal of comprehensive argumentative analysis, laying the groundwork for future exploration of more complex patterns, as well as the incorporation of contextual features in detecting argumentative patterns. Furthermore, our models can support humans in locating argumentation in financial contexts (van der Meer et al., 2024), with potential applications in areas such as investor relations, corporate communication, and financial analysis.

2 Related Work

2.1 Text segmentation in ECCs

In this study, the ROCOI pattern is extracted from Earnings Conference Calls (ECCs): teleconferences that listed companies hold at the presence of financial analysts, following the publication of quarterly results (Givoly and Lakonishok, 1980; Keith and Stent, 2019). In the Q&A session of ECCs, each analyst typically only has one turn for their questions. As a consequence, they adopt an idiosyncratic question-compression strategy whereby multiple questions on different topics are asked within a single turn before any response. These topically homogeneous sequences of utterances that compose the multi-issue question turns are called Maximal Interrogative Units (MIUs) (D'Agostino et al., 2024a,b). MIUs are the units of analysis within which argumentative patterns–in this case, ROCOIs–are identified and extracted.

2.2 The Request of Confirmation of Inference: An argumentative pattern in ECCs

A Request of Confirmation of Inference (ROCOI, previously introduced and qualitatively studied by Rocci and Raimondo (2018)) is an argumentative pattern in ECCs that is originated in MIUs. It is *relevant* to the discussion in the sense that it creates an argumentative confrontation (van Eemeren and Grootendorst, 2004).

A ROCOI is an assertive question, i.e., in which a stance is asserted by the questioner. As a consequence, when it is formulated directly, a ROCOI is a closed question. Moreover, ROCOIs make explicit by lexical means the fact that the stance asserted is the result of an inferential process, the conclusion of which is expected to be (dis)confirmed by the interlocutor. This results in the ROCOI being a challenging question, regardless of the degree of semantic indirectness of its formulation.

Example 1 shows some ROCOIs; in bold, the lexical elements that indicate the inferential process, which constitutes the keystone of the pattern.

(1) a. **Does that mean** that customers are reluctant to term out these sort of prices?
b. **Just wondering**, are you seeing supply opening up in urban areas?
c. **Should we think** of the capital commitment has a hard cap now.
d. **Is it fair to say** that you've maxed out on what was pre-approved at the AGM and that any incremental issue from here would require AGM approval?

Previous studies on the ROCOI (Rocci and Raimondo, 2018; D'Agostino and Rocci, 2024) identify subcategories of the pattern. This article considers the class that D'Agostino and Rocci (2024) call Type 1, that is, ROCOIs in which the inferential conclusion–of which the questioner asks confirmation–is part of the interrogative sentence, as shown in all questions of Example 1. The reason behind the choice is twofold: on the one hand, Type 1 ROCOIs are more compact, in the sense that the conclusion and the question pertain to the same unit, and are therefore more easily identifiable; on the other hand, they are the most frequent ones.

3 Method

We outline the dataset, task formulation, and evaluation setup for the ROCOI extraction approaches.

3.1 Dataset

Our work focuses on a dataset that comprises 60 Earning Conference Calls (ECCs) between 2020–2023 for companies Airbnb (ABNB), British Petroleum (BP), Credit Suisse (CS), Door Dash (DASH), Hasbro (HAS), Shell (SHEL), Exxon Mobil (XOM) and Zillow (Z), for a total of 1377 MIUs. Manual annotation resulted in 180 MIUs featuring ROCOIs, in total containing 193 unique ROCOI patterns. Of these, 134 were Type 1 ROCOIs, and thus represent the final corpus for this study. The annotation was first carried out by trained annotators; student assistants hired specifically for the annotation procedure. Each document was analysed by two to four annotators in variable configuration. The resulting pairwise agreement on the argumentative pattern annotation task (for which the ROCOI is one of eight possible values) is moderate to substantial, with a Cohen's kappa (Cohen, 1960) value ranging from κ=0.41 to κ=0.76. Further information about the annotation guidelines is provided in Lucchini and D'Agostino (2023)[1]. In total, 18% of tokens in the dataset are part of a ROCOI, whereas 82% of tokens are non-ROCOI tokens.

3.2 Task formulation

We compare two task formulations for ROCOI extraction: (1) sequence labeling and (2) text generation. These two tasks allow us to compare the results obtained from applying an *extractive* and an *abstractive* paradigm. Extraction, where we mark the boundaries between the presence and absence of a ROCOI, represents the standard method of identifying a substructure. However, such an approach usually requires ample training data. In contrast, abstraction, which involves generating the part of the input text that contains the ROCOI pattern, is similar to more recent state-of-the-art LLMs. We aim to investigate which approach works better given our relatively small dataset. We describe each task formulation separately and provide extensive details about hyperparameters and training settings for all models in Appendix A.

(1) Sequence labeling In sequence labeling for ROCOI extraction, IOB labeling consists in the tagging of tokenized sequences, indicating for each token whether it does not pertain to the desired sequence (tag: "O"), it is the first token of the sequence (tag: "B"), or it is an inner token of the sequence after the first one (tag: "I"); the padding tokens are assigned a default system-ignored tag "-100". This tagging format is often employed for Named Entity Recognition (NER) tasks, and therefore the "B" and "I" tags typically further indicate to what category the tagged entity belongs to (e.g., person, location, etc.). This study only considers one type of pattern and thus does not employ further class specifications per class.

We experiment with 5 open-source models in total; three of those are encoder-only models:

TinyBERT The smallest model to gauge task complexity. If the smallest model can learn it well, we do not need to train a more capable model (Jiao et al., 2020).

Vanilla BERT Since it is commonly used as a baseline (Devlin et al., 2019).

SpanBERT As a version of BERT that is optimized to represent spans of text, since ROCOIs are often single contiguous spans (Joshi et al., 2020).

In addition, we also experiment with two encoder-decoder models:

T5 Strong empirical results indicate that this model may be used across contexts and tasks (Raffel et al., 2020).

FlanT5 Updated version of T5 that includes a wider array of tasks, the model may generalize better to unseen tasks (Longpre et al., 2023).

(2) Text-to-text generation For this task, the pattern is considered a substring of the MIU given as an input; hence, the output corresponds to a *verbatim* generation of a portion of the wider unit (similar to the use of the text-to-text architecture already intended by Raffel et al. (2020)). Therefore, particular attention must be devoted to the quality of the generation and, specifically, that the fine-tuned model (i) does report the exact portion of the string that contains the pattern and not an adaptation (such as, for instance, a summarization of the original text) (ii) does learn that a pattern is a continuous sequence within the text and (iii) does not repeat multiple subpatterns (whether correct or not) to fill the plausible length of the expected pattern.

This portion of the study is carried out on two text-to-text model families:

[1]The dataset is available on GitHub: [ADDRESS REDACTED].

BART serves as the encoder-decoder counterpart to our BERT baseline for sequence labeling. We use the base and large varieties (Lewis et al., 2020) to further investigate the impact of model size.

T5 in the small, base, and large varieties, again to see whether a more versatile text-to-text training procedure benefits performance (Raffel et al., 2020).

3.3 Evaluation

We outline how we evaluate models on each task formulation.

3.3.1 Sequence labeling

We initially aimed to adopt a similar evaluation approach as Named Entity Recognition (NER), as it shares the IOB tagging setup (Li et al., 2020). Performance in NER and similar tasks is traditionally evaluated at the **token level** (Tjong Kim Sang and Buchholz, 2000). However, tagging is typically performed (a) on short sequences, (b) in multiclass classification, and (c) featuring multiple units in a text; none of these characteristics strictly hold for ROCOIs. Even in the NER extraction domain, however, there has been a propensity towards evaluation at the full entity level, especially if the prediction is aimed at downstream tasks (Segura-Bedmar et al., 2013). Since ROCOIs are long and complex spans of text with potentially variable boundaries, we additionally adopt **span-level evaluation** and compare it to individual token-level evaluation.

Token-level evaluation At the token level, we first provide an overview of the accuracy in the prediction by individual tags ('O', 'I', 'B'). Then we aggregate the tags and provide a measure of precision, recall, and F1 score, alongside the calculation of token-based Krippendorff's α (Krippendorff, 1970).

Span-level evaluation To evaluate the entire span over which the ROCOI develops and not only the individual tokens that constitute it, we make use of the ROUGE-L metric, to determine the longest matching string, as well as the Gamma (Γ) method for inter-annotator agreement measure and alignment (Mathet et al., 2015)[2] in a basic, one-label, positional dissimilarity detection configuration.

3.3.2 Text-to-text generation

For the text-to-text generation evaluation, we use various metrics to investigate the quality of the extracted pattern. Each model is evaluated according to six metrics, clustered into three classes, each of which corresponds to a different way of interpreting the nature of the task: *syntactic* (pattern matching), *semantic* (embedding similarity), or *annotation* (inter-annotator agreement). The rationale behind such a three-fold choice lies in the nature of generative models: on the one hand, they tend to be too creative despite being prompted to extract verbatim text. This would not be captured by semantic metrics but is counterbalanced by syntactic metrics. On the other hand, syntactic evaluation cannot capture whether some slightly shifted boundary still correctly identifies the core of the pattern–which can however be reintegrated into the equation to some extent by the use of semantic similarity (although not entirely, since such metrics are not specialized in ROCOI *core meaning* detection, similar to sequence labeling). Inter-annotator agreement metris works as a sanity check that decidedly signals the presence of ill-formed sequences in generated patterns.

Syntactic evaluation In this view, the extraction performance is evaluated in terms of string matching. The first naïve evaluation that establishes the baseline consists of checking whether the pattern is present in the extracted string. We call this evaluation "pattern matching" and its most obvious flaws are that (a) over-extraction to the point of reporting the entire original string is a hit and (b) even slight under-extraction is a complete miss. The three possible values are 'full match' (if the retrieved string contains exactly the correct pattern), 'partial match' (if the retrieved string contains at least the full correct pattern), and 'no match' otherwise; reported are the frequency distributions across the three classes. This is paired with a more refined version of such an evaluation, that is, the calculation of the ROUGE score (Lin, 2004); specifically the ROUGE-L metric, which identifies the longest co-occurring sequence.

Semantic evaluation In this case, what is evaluated is the semantic distance between the predicted and the actual pattern. This is achieved by (1) calculating a simple Euclidean distance between the embedding representation of the patterns and (2) applying some well-established evaluation methods that are typically used for text generation and summarization: notably (a) BERTScore (Zhang et al., 2020) and (b) Sentence-BERT (SBERT) (Reimers and Gurevych, 2019).

[2]Taken from the Python library `pygamma-agreement` (https://github.com/bootphon/pygamma-agreement)

Model	Accuracy		
	O	I	B
BERT (base)	0.93	0.61	**0.70**
TinyBERT	0.92	0.39	0.40
SpanBERT	0.89	0.61	0.60
T5 (base)	**0.95**	**0.67**	0.65
FlanT5 (base)	0.92	**0.67**	**0.70**

Table 1: Sequence labeling accuracy by tag. The best models are shown in bold, second best underlined.

Annotation agreement evaluation The true pattern can be considered a gold standard annotation and the extracted pattern a machine-generated annotation; in this perspective, the two are compared with a tool designed to capture the inter-annotator agreement and the dissimilarity in span boundaries. In particular, we use the Gamma (Γ) method for inter-annotator agreement measure and alignment (Mathet et al., 2015). The metric cannot compute on instances in which the extracted pattern is not a lexical match to a substring of the input text, and thus tells us that the generated string is ill-formed.

4 Results and discussion

We describe our results after training the models on the two tasks: sequence labeling and text-to-text generation respectively.

4.1 Sequence labeling

Table 1 reports the accuracy values by individual tag. As reported in Section 3.1, 82% of tokens in the dataset are non-ROCOI elements; these are identified by 'O' tags. Therefore, since they represent the most frequent type, as expected 'O' tokens reach a higher accuracy across models. On the contrary, 'B'-type tokens understandably are the least frequent ones in the corpus but its accuracy levels are not far from that of 'I' tokens overall – if not better. It is worth noticing that SpanBERT appears to be performing badly despite being optimized for encoding contiguous spans of texts. It achieves the lowest accuracy on the 'O' tag, indicating it most strongly mislocates ROCOI patterns in the text. At this stage, the best performing models seem to be the two belonging to the T5 family (both best in two out of three accuracy values), followed by vanilla BERT (second best in two out of three accuracy values).

Further classification results aggregated over the three tag categories are displayed in Table 2, both at the token level (former four columns) and span level (latter two columns). Token-level evaluation appears to favor FlanT5, which achieves the highest results in three out of four metrics and is second best in the remaining one. Surprisingly, SpanBERT performs below par in full span detection, according to span-level evaluation results, which are instead dominated again by T5 (ROUGE-L = 0.90) and FlanT5 ($\Gamma = 0.63$).

Earlier, we reported the annotation agreement among human annotators in terms of Cohen's κ (see Section 3.1). As we're working with a single category, however, the metric is roughly comparable with Krippendorff's α. This allows us to conclude that all models, except for TinyBERT, achieve an IAA performance within the human range ($0.41 \leq \kappa \leq 0.76$). This indicates that we may use automatic ROCOI extraction for machine annotation for new samples in the future. However, the machine annotations fail in a way that is not captured by this metric, or disagree with human annotators in novel ways. Hence, we set out to further understand the limitations of the automatic ROCOI extraction approach in Section 5.1.

4.2 Text-to-text generation

We present the evaluation results sorted by evaluation approach type (*syntactic*, *semantic*, *annotation*), each of which is presented in a dedicated table.

Table 3 reports *syntactic* evaluation. For both evaluation methods, the two BART models appear to be by far the best-performing ones, particularly the *large* configuration – with best results across all metrics. *Semantic* measures are reported in Table 4. The baseline metric represented by raw Euclidean distance between the true and predicted pattern favors BART models; moreover, both SBERT and BERTScore, again identify BART-large as the best-performing model, reaching $F1 = 0.94$. Similar outcomes are shown in Table 5, which displays surprisingly bad results for the T5 models on the *inter-annotator agreement* metrics. This will be appropriately discussed in Section 5.2.

Different metrics capture different aspects of the ROCOI extraction task in a text-to-text generation setup, For instance, syntactic pattern matching informs us of the capability to lexically overlap with the ground truth patterns, while semantic evaluation allows us to observe how well the model captures the underlying meaning and intent of the ROCOI spans. We observe that BART models achieve

Model	Token-level Precision	Recall	F1	α	Span-level ROUGE-L	Γ
BERT (base)	0.22	0.25	0.23	0.58	0.87	0.49
TinyBERT	0.09	0.05	0.06	0.37	0.82	0.60
SpanBERT	0.27	**0.30**	0.29	0.51	0.83	0.47
T5 (base)	0.17	0.15	0.16	**0.67**	**0.90**	0.56
FlanT5 (base)	**0.32**	**0.30**	**0.31**	0.61	0.87	**0.63**

Table 2: Additional results for the sequence labeling approaches. The best models are shown in bold, second best underlined.

good performance along all three dimensions for this task.

5 Error analysis

In addition to our previous results, we present a qualitative analysis of the predicted patterns using the best-performing models in both tasks. Specifically, we observe the onset point and length of all extracted patterns in the test set to identify whether models tend to make consistent mistakes. Further, for the sequence labeling task, we also present an overview of the distribution of ill-formed sequences in the prediction, that is, cases in which a sequence onset is not correctly followed by the next element in the sequence: a 'B' tag immediately followed by an 'O' (not possible in a well-formed ROCOI). While the results here summarize the findings, Tables 9 and 10 for sequence labeling and text generation respectively–available in Appendix B–report by row the measures over each instance in the test set. Shown there are the offsets of the prediction with respect to the true pattern for both the beginning of the predicted pattern and its length – calculated in terms of token numbers – and the absolute number of ill-formed sequences in the prediction for sequence labeling.

5.1 Sequence labeling

Concerning BERT, the start of the predicted pattern is correctly aligned in 55% of cases, too early in 20% of cases, and too late only in 15% of cases. The pattern was not found in the last 10% of test instances. In terms of length, the exact right length is extracted in 15% of cases, while it typically tends to extract patterns too short (45% of cases); the extracted pattern is too long in 30% of cases, which are all below or equal to 5 tokens of difference from the gold standard. These results are accompanied by the observation that in 25% of test instances, some ill-formed sequences are present in the prediction; however, never more than two per instance.

As for T5, perfect alignment with the start of the pattern occurs in 60% of cases, whereas both early and late onset constitute 10% of cases. For this model, the pattern was not found in the unit in 20% of cases. The exact length is extracted in 15% of cases, while the majority of predicted patterns appear to be, also in this case, shorter than expected (55%) – of which half are below 10 tokens of distance from the true length. In 10% of cases, the extracted pattern is longer than the gold standard, always by 3 tokens. Worth noting is the near-perfect acquisition of the IOB-tagging rules, which is reflected in a single instance of ill-formed sequence; this is moreover associated with an instance of non-extraction of the pattern.

FlanT5 outperforms both BERT and T5, which is reflected in its predictive capabilities. The right starting point is detected in 70% of cases, while it is too early in 5% of cases and too late in 15%. Extraction of exact right length spurts to 30%, whereas short and long sequences represent 45% and 15% respectively. The pattern is not found in 10% of instances. However, 100% of predicted patterns contain ill-formed sequences, 1 to 4 per instance (mode $= 2$). This is an issue when such an extraction step is integrated into a pipeline–with the concrete risk of error propagation.

Following, a test instance failed by all three models (in bold the ROCOI): "And secondly, on U.S. gas, you're very well-positioned with I believe pretty much fully hedged production for this year, but **I'm wondering if at $2 per MCF gas, you're actually starting to see the opportunity to perhaps take away some of the rigs** and refocus them in the Permian where you keep strongly growing the activity. Thank you." In this example, FlanT5 recognizes three starting points (underlined the tokens corresponding to a 'B' tag in the predicted

Model	Pattern matching			ROUGE-L
	Full match	Partial match	No match	
BART (base)	**0.20**	0.50	0.30	0.63
BART (large)	**0.20**	**0.60**	**0.20**	**0.67**
T5 (small)	0.00	0.45	0.55	0.43
T5 (base)	0.15	0.50	0.35	0.54
T5 (large)	0.00	0.15	0.85	0.31

Table 3: Syntactic evaluation for text-to-text generation. For pattern matching, results must be read as "the higher the better" for full and partial match, and "the lower the better" for no match. The best models are shown in bold, second best underlined.

Model	Euclidean distance	SBERT similarity	BERTScore		
			Precision	Recall	F1
BART (base)	**0.42**	0.07	0.91	0.95	0.93
BART (large)	0.46	**0.08**	**0.92**	**0.96**	**0.94**
T5 (small)	0.46	0.05	0.86	0.93	0.90
T5 (base)	0.59	0.06	0.89	0.94	0.91
T5 (large)	0.54	0.07	0.84	0.90	0.87

Table 4: Semantic evaluation for text-to-text generation. The best models are shown in bold, second best underlined.

Model	Γ
BART (base)	**0.56**
BART (large)	0.54
T5 (small)	0.07
T5 (base)	0.26
T5 (large)	—

Table 5: Annotation agreement evaluation for text-to-text generation. The best models are shown in bold, second best underlined.

sequence) and one well-formed sequence roughly corresponding to the true pattern (in bold the tokens corresponding to 'I' tags): "And secondly, on U.S. gas, you're very well-positioned with I believe pretty much fully hedged production for this year, but I **'m wondering if at $2 per MCF gas, you're actually starting to see the opportunity to perhaps take away some of the rigs and refocus them in the** Permian where you keep strongly growing the activity. Thank you.". FlanT5 therefore not only marks multiple onset points, but some of them may also interrupt ongoing sequences.

Finally, Figure 2 provides a graph overview of the lengths of the ROCOI patterns, both for the true patterns and the predicted spans from each sequence labeling model. An immediate observation is that BERT greatly overestimates the amount of tokens in the pattern. In conclusion, T5 is the most reliable model for onset position prediction (offset mean $= 0.93$) whereas FlanT5 is the best at predicting pattern length (offset mean $= -6.2$), as confirmed by similar length distribution in Figure 2f compared to the gold standard of Figure 2a.

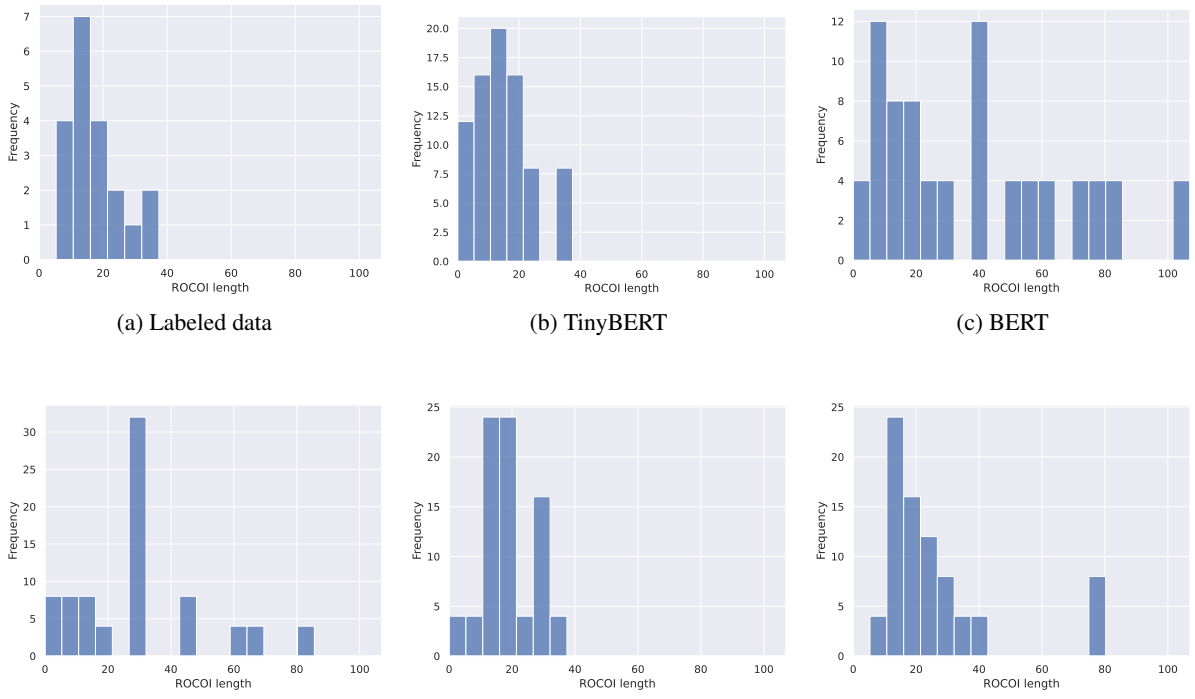
5.2 Text-to-text generation

Qualitative analysis of the text-to-text generation task was conducted for the two varieties of BART models, as they were the best performing across metrics. Understandably, they show similar behavior and the *large* configuration mostly hits some of the misses of the *base* configuration (cf. Table 10).

The right starting point is detected in 45% of cases by BART base, increasing to 60% for BART large. An early detected pattern onset represents 35% of cases for BART base, decreased to 20% for BART large. In both configurations, the detected pattern started late in 5% of cases and the pattern was not detected at all in 15% of cases. The distribution of predicted lengths was the same across both varieties (20% correct, 65% long, 0% short), presumably meaning that already the *base* configuration is powerful enough to pick up such a feature to the best that this model family allows given the quantity of training data available.

In conclusion, both BART models learned to identify the start of the pattern in the vast majority of cases; remaining errors, however, greatly diverge from the gold standard. Unfortunately, both tend to overgenerate in the majority of cases, by a considerable extent (77 tokens on average for BART base, 73 for BART large).

An extreme case is represented by T5-large generation: despite all the safeguards, none of the re-



(a) Labeled data (b) TinyBERT (c) BERT

(d) SpanBERT (e) T5 (f) FlanT5

Figure 2: True (upper left) and predicted (others) ROCOI lengths.

trieved patterns corresponds to a substring of the original text–hence hindering the calculation of the Γ metric in Table 5. For example, compared to the true pattern "Are you suggesting that you could potentially ship to Russia later this year?", the corresponding generation reads: "- And then my follow, as it is in terms of Europe. I just want to clarify that? So this has the potential risk from Russia for approximately 100 million.".

6 Conclusions

This paper introduces a prototypical argumentative pattern that originates in the questions asked during the Q&A sessions of financial dialogues, called the Request Of Confirmation Of Inference (ROCOI). Since argumentation is a pivotal aspect of human communication, the identification and extraction of argumentative patterns is argued to be fundamental in the study of language in interaction. Particularly, given that the identification of argumentative patterns is a challenging yet doable task for trained humans, this study seeks to answer the question of whether language models can perform this task as well.

We adopted two concurrent ML approaches to the extraction of ROCOIs from a wider interrogative unit: sequence labeling and text-to-text generation. Sequence labeling was performed comparing three encoder-only models to two encoder-decoder models; text-to-text generation compared five encoder-decoder models. The models, fine-tuned for the task, were selected due to their nature: they are relatively small open-source models. The sequence labeling approach, evaluated both at the token- and span-level, shows that FlanT5 is the best-performing model. Qualitative observation of the results, however, marks its outputs as potentially unreliable. T5 is therefore the best-performing model both for accuracy and reliability of the output. The text-to-text generation approach identifies BART-large as the best-performing model across syntactic, semantic, and annotation agreement evaluation measures.

In conclusion, this task can be carried out by language models. At the present stage, results suggest that sequence labeling is still the most trustworthy method to approach the task. While results would improve with a larger training dataset, gathering additional samples containing ROCOIs is difficult due to their infrequency. Further work may include the insertion of intermediate steps to fine-tune for similar tasks (such as argumentative sequence labeling) before applying them to ROCOI extraction (van der Meer et al., 2022).

7 Limitations

Our work has several important limitations to consider. While we carefully selected the models that are open source and accepted baselines among related work in Argument Mining literature, our choice of model architecture remains limited. Future work can benefit from investigating how larger (decoder-only) approaches, for instance, those using In-Context Learning, perform on the ROCOI extraction task. Further, our relatively limited dataset size affects the generalizability of our results, especially in cases of context shift. Training models with more data, or increasing the size of the evaluation set may paint a different image of the relative performance among models. Lastly, despite using fixed model checkpoints and consistent dataset splits, we observed that T5's generation outputs exhibit high predictive variability, introducing some uncertainty in our results. In addition, we found that FlanT5 has a systematic tendency to overpredict multiple ROCOI spans within individual samples, potentially inflating certain metrics.

8 Ethical Considerations

Recognizing argumentative content can be biased to the content of the training set. This may result in predictions that are poor in novel contexts or edge cases. Responsible implementations of an extraction system, especially in the financial domain, should always be checked by a human. Our work is a first attempt at creating a system for analyzing argumentative patterns for financial dialogues. Situating our approach in an ecosystem that contains checks and balances will not only ensure responsible use of the predictive model but also may yield valuable insights into the actual use of the model.

References

Jianzhu Bao, Chuang Fan, Jipeng Wu, Yixue Dang, Jiachen Du, and Ruifeng Xu. 2021. A neural transition-based model for argumentation mining. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6354–6364, Online. Association for Computational Linguistics.

Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Giulia D'Agostino, Chris Reed, and Daniele Puccinelli. 2024a. Segmentation of Complex Question Turns for Argument Mining: A Corpus-based Study in the Financial Domain. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14524–14530, Torino, Italia. ELRA and ICCL.

Giulia D'Agostino and Andrea Rocci. 2024. Argumentative patterns in the context of dialogical exchanges in the financial domain. In *Proceedings of the 24th Edition of the Workshop on Computational Models of Natural Argument (CMNA 24)*, Hagen, Germany.

Giulia D'Agostino, Ella Schad, Eimar Maguire, Costanza Lucchini, Andrea Rocci, and Chris Reed. 2024b. Superquestions and some ways to answer them. *Journal of Argumentation in Context*. In press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11–22, Vancouver, Canada. Association for Computational Linguistics.

Debanjan Ghosh, Smaranda Muresan, Nina Wacholder, Mark Aakhus, and Matthew Mitsui. 2014. Analyzing argumentative discourse units in online interactions. In *Proceedings of the first workshop on argumentation mining*, pages 39–48.

Dan Givoly and Josef Lakonishok. 1980. Financial analysts' forecasts of earnings: Their value to investors. *Journal of Banking & Finance*, 4(3):221–233.

Deniz Gorur, Antonio Rago, and Francesca Toni. 2024. Can large language models perform relation-based argument mining? *arXiv preprint arXiv:2402.11243*.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the association for computational linguistics*, 8:64–77.

Katherine Keith and Amanda Stent. 2019. Modeling financial analysts' decision making via the pragmatics and semantics of earnings calls. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 493–503, Florence, Italy. Association for Computational Linguistics.

Klaus Krippendorff. 1970. Estimating the Reliability, Systematic Error and Random Error of Interval Data. *Educational and Psychological Measurement*, 30(1):61–70.

Tatsuki Kuribayashi, Hiroki Ouchi, Naoya Inoue, Paul Reisert, Toshinori Miyoshi, Jun Suzuki, and Kentaro Inui. 2019. An empirical study of span representations in argumentation structure parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4691–4698, Florence, Italy. Association for Computational Linguistics.

John Lawrence and Chris Reed. 2019. Argument Mining: A Survey. *Computational Linguistics*, 45(4):765–818.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2020. A survey on deep learning for named entity recognition. *IEEE transactions on knowledge and data engineering*, 34(1):50–70.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR.

Costanza Lucchini and Giulia D'Agostino. 2023. Good answers, better questions. Building an annotation scheme for financial dialogues. Technical report.

Fabrizio Macagno and Douglas Walton. 2014. Argumentation schemes and topical relations. In Giovanni Gobber and Andrea Rocci, editors, *Language, reason and education*, pages 185–216. Peter Lang, Bern.

Yann Mathet, Antoine Widlöcher, and Jean-Philippe Métivier. 2015. The Unified and Holistic Method Gamma (Γ) for Inter-Annotator Agreement Measure and Alignment. *Computational Linguistics*, 41(3):437–479.

Andreas Peldszus and Manfred Stede. 2013. From Argument Diagrams to Argumentation Mining in Texts. *International Journal of Cognitive Informatics and Natural Intelligence*, 7(1):1–31.

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):1–67.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Andrea Rocci and Carlo Raimondo. 2018. Dialogical Argumentation in Financial Conference Calls: The Request of Confirmation of Inference (ROCOI). In *Argumentation and Inference: Proceedings of the 2nd European Conference on Argumentation*, pages 699–715. College Publications.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 Task 9 : Extraction of Drug-Drug Interactions from Biomedical Texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 Shared Task Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.

Michiel van der Meer, Enrico Liscio, Catholijn Jonker, Aske Plaat, Piek Vossen, and Pradeep Murukannaiah. 2024. A hybrid intelligence method for argument mining. *Journal of Artificial Intelligence Research*, 80:1187–1222.

Michiel van der Meer, Myrthe Reuver, Urja Khurana, Lea Krause, and Selene Báez Santamaría. 2022. Will it blend? mixing training paradigms & prompting for argument quality prediction. In *Proceedings of the 9th Workshop on Argument Mining*, pages 95–103.

Frans van Eemeren and Rob Grootendorst. 2004. *A Systematic Theory of Argumentation: The Pragma-dialectical Approach*. Cambridge University Press.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Experimental details

A.1 Training parameters

We present additional details regarding the usage of pretrained models for the two formulations of the ROCOI extraction. We present an overview of the initial model checkpoints and their parameter counts in Table 6. The hyperparameters to train the models on the sequence labeling task are given in Table 7, and the ones for text-to-text generation are given in Table 8. Training a single model generally takes up to one hour at most on modern hardware (one RTX3090 or A100 GPU).

Sequence labeling For the sequence labeling models, we train on the training set (75% of total available samples) while observing metrics on a validation set (10% of samples). We pick the model iteration with the highest token-level F_1 score and evaluate that model on the test set (15% of samples) to obtain the results reported in Tables 1 and 2. We use the same split for each experiment.

Text-to-text sequence generation For the text-to-text generation models, we train on the training set (75% of total available samples) while observing metrics on a validation set (10% of samples). We optimized hyperparameters and picked the best model iteration with the lowest loss value, and evaluated that model on the test set (15% of samples) to obtain the results reported in Tables 3, 4, and 5. We use the same split for each experiment.

B Error analysis

We present additional details upon which we based our qualitative observations of Section 5. Particularly, we display the the raw numerical data for each test instance, which in the body of the paper was instead merged in the form of percentage over the total. Table 9 refers to the sequence labeling task and reports begin- and length- offsets of the predicted patterns with respect to the gold standard, alongside the number of ill-formed sequences in the tag sequence. Table 10 presents begin- and length- offset numbers only, from the text-to-text generation task.

Model	Checkpoint	Size
BERT (base)	`google-bert/bert-base-uncased`	109M
SpanBERT	`SpanBERT/spanbert-base-cased`	108M
TinyBERT	`huawei-noah/TinyBERT_General_4L_312D`	14M
T5 (base)	`google-t5/t5-base`	110M
Flan-T5 (base)	`google-t5/flan-t5-base`	110M
BART (base)	`facebook/bart-base`	139M
BART (large)	`facebook/bart-large`	406M
T5 (small)	`google-t5/t5-small`	61M
T5 (base)	`google-t5/t5-base`	223M
T5 (large)	`google-t5/t5-large`	738M

Table 6: Description of each model and the specific checkpoint we used.

Model	Parameter	Value
BERT (base)	learning rate	2e-05
SpanBERT	learning rate	2e-05
TinyBERT	learning rate	2e-05
T5 (base)	learning rate	4e-04
Flan-T5 (base)	learning rate	4e-04
all	batch size	16
all	max sequence length	256
all	max epochs	100

Table 7: Hyperparameters for the sequence labeling approaches.

Model	Parameter	Value
all	learning rate	6e-06
BART *all*	batch size	4
T5 *(all)*	batch size	6
all	max sequence length	256
all	max epochs	100

Table 8: Hyperparameters for the text-to-text approaches.

begin offset	length offset	ill-formed sequences
0	0	0
0	-4	0
0	-6	0
0	0	0
181	-16	1
n.a.	n.a	0
0	4	0
0	-33	0
0	0	0
-33	3	1
49	-14	0
n.a.	n.a	0
0	-30	0
-40	5	0
0	1	2
0	-27	1
0	13	1
-1	2	0
-17	-6	0
0	-27	0

(a) BERT (base)

begin offset	length offset	ill-formed sequences
0	-2	0
n.a.	n.a.	0
0	0	0
0	0	0
n.a.	n.a.	1
0	-5	0
0	-9	0
0	-33	0
0	-2	0
0	0	0
-41	-16	0
n.a.	n.a.	0
0	-34	0
n.a.	n.a.	0
0	3	0
1	-41	0
0	3	0
73	-5	0
-18	-6	0
0	-15	0

(b) T5 (base)

begin offset	length offset	ill-formed sequences
0	-1	4
0	0	4
0	0	3
0	0	1
0	10	2
0	0	3
0	4	2
0	-33	3
0	0	2
0	0	3
n.a.	n.a.	2
0	-1	2
0	-34	3
n.a.	n.a.	2
0	5	3
1	-14	3
12	-5	2
69	-1	2
-18	-6	2
0	-26	4

(c) FlanT5 (base)

Table 9: Qualitative error analysis: sequence labeling approach. Reported the three best performing models. For each sub-table, the first two columns indicate offsets (predicted-true) and the third one indicates the absolute number of instances. The best value is zero for all features.

begin offset	length offset
-218	187
0	0
n.a.	n.a.
0	0
0	78
158	5
0	71
-47	47
0	0
-160	77
-179	34
-196	196
0	0
n.a.	n.a.
0	33
n.a.	n.a.
0	70
-4	87
-74	74
0	38

(a) BART (base)

begin offset	length offset
0	182
0	228
n.a.	n.a.
0	0
0	78
158	5
0	22
-47	47
0	0
0	35
-179	34
0	0
0	0
-186	85
0	33
n.a.	n.a.
0	70
-4	87
n.a.	n.a.
0	38

(b) BART (large)

Table 10: Qualitative error analysis: text-to-text sequence generation approach. Reported the two best performing models. For each sub-table, the two columns indicate offsets (predicted-true). The best value is zero for all features.