Speech Act Patterns for Improving Generalizability of Explainable Politeness Detection Models

Ahmad Aljanaideh
Bentley University
Waltham, MA, USA
aaljanaideh@bentley.edu

Abstract

The lack of explainability in state-of-the-art Natural Language Understanding (NLU) classification models has increased interest in developing techniques for improving explainable linear feature-based models (e.g., Logistic Regression/SVM). Politeness detection is a task that exemplifies this interest. While those techniques perform well on the task when applied to data from the same domain as the training data, they lack generalizability and thus fall short when applied to data from other domains. This is due to their reliance on discovering domain-specific word-level features. We introduce a method for improving the generalizability of explainable politeness models by relying on speech act patterns instead of words, leveraging speech act labels assigned by the GPT-4 model. This approach goes beyond the mere words and injects intent into politeness classification models, enhancing their generalizability. Results demonstrate that the proposed method achieves state-of-the-art accuracy in the cross-domain setting among explainable methods, while falling short in the in-domain setting. Our findings illustrate that explainable models can benefit from Large Language Models.

1 Introduction

While recent NLU classification techniques such as BERT (i.a. Devlin et al., 2018; Liu, 2019; He et al., 2020) and the GPT series (i.a. Radford, 2018; Radford et al., 2019; Brown, 2020; Achiam et al., 2023) achieve cutting-edge accuracy, their complexity hinders their ability to explain their decisions. This has led to growing interest in developing techniques for improving explainable linear models (e.g., Logistic Regression/SVM). Explainable models are essential for transparent adoption of AI (i.a. Clement et al., 2023; Chen et al., 2024; Ali et al., 2023; Hassija et al., 2024; Linardatos et al., 2020).

An example task for which practitioners developed explainable approaches is politeness detection in text. Recent approaches for improving explainable models for this task (Aljanaideh et al., 2020) rely on discovering fine-grained context patterns of words from a corpus annotated for politeness, and then using the resulting patterns as features to train an explainable model, allowing examination of model coefficients to understand its decision-making process. While these features help reduce the gap in performance between explainable models and more complex ones such as BERT (Devlin et al., 2018) when applied to data from the same domain as the training data, they fall short when applied to data from other domains. This is due to their reliance on domain-specific word-level features.

We introduce a novel method for improving the generalizability of explainable politeness detection models. The method relies on clustering speech act (e.g. *gratitude*) textual elements (words, phrases and sentences), and using the resulting clusters as features to improve the performance of explainable models for the task. This allows going beyond words to include the intent of language use. GPT-4 (Achiam et al., 2023) is used to assign speech act categories to textual elements, and element embeddings are then clustered using Aljanaideh et al. (2020)'s algorithm to obtain interpretable features an explainable model can be trained with.

Results show the proposed method achieves state-of-the-art accuracy on the task of explainable politeness detection in the cross-domain setting, helping reduce the gap in performance between explainable models and a fine-tuned BERT (Devlin et al., 2018), while falling short in the in-domain setting. The method also automatically discovers subtle politeness cues (e.g., apologizing statements such as *I should not have used rollback.*). Our findings indicate that explainable AI models can be improved by leveraging LLMs.

Findings of the Association for Computational Linguistics: ACL 2025, pages 18945–18954

2 Proposed Method

Our goal is to improve the performance of explainable linear models (e.g., Logistic Regression/SVM) for politeness detection. Aljanaideh et al. (2020) achieves this by discovering fine-grained context patterns of words using clustering of word embeddings. The resulting clusters are then used as features to train an explainable model for the task. In our method, we rely on speech acts instead of words. Specifically, we discover fine-grained clusters of speech acts, and then use those clusters as features to train an explainable model for the task. First, we prompt GPT-4 to extract textual elements (words, phrases, and sentences) associated with pre-defined speech act categories. We then cluster embeddings of each speech act element category using Aljanaideh et al. (2020)'s algorithm. This helps discover different ways of using speech acts in (im)polite contexts, even when there is no syntactic overlap between the different elements. For example, *I am sorry* and *apologies* are assigned to the same cluster despite having no words in common, since they both contain a form of apology. We describe each of those steps below.

2.1 Speech Acts Labeling

We select key speech act categories relevant to the task and use an LLM to extract textual elements (word, phrase, or sentence) associated with those categories. A speech act is the action encompassed in words expressed by an individual (Austin, 1975). Speech Act Tagging (also known as Dialogue Act Tagging) is a long-standing task in NLU that aims to classify utterances in dialogue based on their function or communicative intent (Stolcke et al., 2000). Existing methods for this task use LLMs (Kim et al., 2024) while others use Deep Learning (i.a. Raheja and Tetreault, 2019; Li et al., 2018). Based on work in politeness literature (Danescu-Niculescu-Mizil et al., 2013), we choose 10 speech act categories listed in Table 1.[1] We then prompt GPT-4 to extract words, phrases, and sentences associated with each category. For example, if the request is *Hello, Could you please help me?*, then the model would assign *Greeting* to *Hello*, and *Request* to *Could you please help me?*. We pass the entire text item as input instead of individual parts since it has been shown that contextual information has been shown to help speech act classification performance (Ribeiro et al., 2019; Lin et al., 2023).[2]

Speech Act Category	Example	%
Gratitude	Thanks very much for...	5
Greeting	Hey there.	6
Apologizing	I am sorry.	4
Direct Question	what's the problem?	39
Direct Start	So how about ...	3
Hedging	This might be ...	18
Positive senti.	Nice photo!	5
Negative senti.	Bad edit.	7
Request	Could you please	11
Deference	Nice work!	1

Table 1: Percentage of each speech act category among textual elements detected by GPT-4 in the training set.

2.2 Clustering

Next, we cluster textual element embeddings of each speech act category. First, we compute an embedding for each speech act textual element (word/phrase/sentence) by averaging its constituent BERT word embeddings. This helps in encoding the element's surrounding context. For example, if the text item is: *Hello,* ***could you please help me?*** *I am having trouble finding the link*, and the word embeddings of the target element (in bold) are averaged. This embedding would encode the surrounding context (*Hello* and *I am having trouble...*). We then use Aljanaideh et al. (2020)'s algorithm to cluster embeddings of elements of each speech act category. Aljanaideh et al. (2020)'s algorithm is a decision tree that clusters occurrences of a word using the contextualized embeddings of the word and the labels of items the word appeared in. In our case, we use the algorithm to cluster speech act elements (words/phrases/sentences) using the embeddings of the elements and the labels of items the elements appeared in. This helps discover (im)polite contexts for using each speech act. The result is a set of clusters for each speech act category. Each cluster contains elements that use the corresponding speech act in a specific context. The clustering is applied on the training set of Wikipedia requests described in the next section.

[1] Appendix A shows how Danescu-Niculescu-Mizil et al. (2013)'s features were used to infer the speech act categories used in this work.

[2] Appendix B provides the prompt we used. In the prompt, we define a "phrase" as being a *any meaningful sequence of words standing together as a conceptual unit*. The "word" and "sentence" definitions are not included in the prompt as they are well-established linguistic concepts.

3 Dataset

We use the Stanford Politeness Corpus (Danescu-Niculescu-Mizil et al., 2013). This corpus contains two sub-datasets of requests: one from the Wikipedia discussion forum where participants discuss article edits, and another from Stack Exchange, a QA forum on various topics (e.g. math, farming). The fact that questions span various topics makes it suitable for cross-domain evaluation of politeness cues. The Wikipedia portion contains 2,176 requests, while the Stack Exchange portion contains 3,302 requests. Each request is assigned 'polite' or 'impolite' based on the average human annotations. We use the same training/development/test split as Aljanaideh et al. (2020).

4 Results

We apply our proposed method to the Stanford Politeness Corpus. Below, we describe the results. First, we report the distribution of speech act categories assigned by GPT-4 to obtain insight into the corpus. We then perform qualitative analysis on a sample of discovered clusters. Finally, we describe a politeness feature-based explainable classifier that uses the discovered clusters as features and compare its accuracy to existing work on politeness.

4.1 GPT-4 Speech Act Analysis

Table 1 shows the distribution of speech act categories assigned by GPT-4 among the elements extracted from the Wikipedia discussion forum request training set. Direct questions, hedging, and requests are the most frequent categories. Categories such as apologizing, gratitude, and greeting are less frequent, but they proved helpful in predicting politeness (Danescu-Niculescu-Mizil et al., 2013). We manually analyzed 100 elements sampled from the training dataset to validate GPT-4 ability in assigning speech acts. The model achieves 83% accuracy. The most common error was assigning *Hedging* to phrases that are closer to being *Direct questions* (e.g., *How could we know if someone verified it?*. On the other hand, we observed GPT-4 generally captures apologizing, gratitude, and greeting very accurately [3].

Table 2 shows examples of speech act clusters discovered with our method. We analyze a sample of those clusters. Direct Questions_1 represents a cluster of element embeddings from polite requests that contained questions (e.g. *What are the sources ...?*) that were asked after a somewhat polite start. In the first example, the speaker starts with hedging (*If you can remember...*), while in the second example, the speaker starts with gratitude (*Thank you*). Direct question_2 represents a cluster of element embeddings from mostly impolite requests which also contained direct questions (e.g., *what other monastery...*) after starting in a direct tone (e.g., *Lionel, tell me the truth...*). Hedging_1 represents a cluster of element embeddings from requests that lean impolite where the speaker starts by describing a situation (e.g., *That's a couple of levels...*) and then follows by a form of hedging (e.g., *But if you just got the idea today...*). Hedging _2 represents a cluster of element embeddings from mostly polite requests where the speaker uses hedging (*If you have time ...*) after starting in a polite tone (e.g., *Hello Roman...*). Interestingly, elements that contain no word overlap (e.g., *so I can refresh my memory* and *If you have time...*) were clustered together. In Aljanaideh et al. (2020)'s method, those two elements wouldn't get assigned to the same cluster due to having no words in common.[4]

4.2 Politeness Classification

We evaluate the predictive power of speech act clusters by using cluster ids as unigram-like features for training explainable (Logistic Regression&SVM) politeness classifiers. Following Aljanaideh et al. (2020), we generate features based on clusters by assigning each speech act element to the closest corresponding cluster, and using the cluster ids as features for the item. Following existing literature, we evaluate the proposed politeness features in both the in-domain (train on Wikipedia and test on Wikipedia) and cross-domain settings (train on Wikipedia and test on Stack Exchange). Table 3 shows the test accuracy for both domains when using Logistic Regression (LR) and Support Vector Machine (SVM) models. Unigram represents context-free words, while Politeness Strategies represent 21 politeness word features from Danescu-Niculescu-Mizil et al. (2013). Clusters (words) represent word cluster features from Aljanaideh et al. (2020). Clusters (Speech Acts) refer to the method described in this work. The last row is a fine-tuned BERT-base model.

Word clusters (Rows #3-5) improve in-domain

[3] More details on the evaluation process results can be found in Appendix C

[4] More cluster analysis are provided in E.

Cluster ID	Items	Polite %	Example	Label
Direct Questions_1	5	100	If you can remember, would you mind giving a bit more information about this photo e.g. **where you took it**?	+
			Thank you for your answer. **What are the sources for region 4 (Australia) release dates**?	+
Direct Questions_2	166	28	In this case the ... **what other monastery with this name exists**?	-
			tell me the truth. **How long have you known BelloMello was wMo**?	-
Hedging_1	172	42	I'm in Outer Mongolia [...]. **Maybe we can meet**?	+
			That's a couple levels of [..] But **if you just got the idea today**, then how can you know the false positive level is[...]?	-
Hedging_2	88	89	I have looked back... **so I can refresh my memory and be best informed what action might be appropriate**?	+
			Hello Roman! **If you have time**, could you add a <url> infobox...	+

Table 2: Example clusters discovered with our method. The columns show the cluster ID, number of items, percentage of polite items, examples, and labels (+ for polite, - for impolite). The bold text highlights the keyword/phrase/sentence detected by GPT-4 in the example.

Row#	Features	In-domain		Cross-domain	
	Model	SVM	LR	SVM	LR
1	Politeness Strategies (Danescu-Niculescu-Mizil et al., 2013)	78.8	76.4	60.3	60.8
2	unigrams + Politeness Strategies	82.2	80.5	67.2	65.5
3	Clusters (words) (Aljanaideh et al., 2020)	84.1	81.7	63.1	66.4
4	Clusters (words) + Politeness Strategies	85.1	82.0	65.9	65.5
5	Clusters (words) + unigrams + Politeness Strategies	83.2	84.1	65.8	65.6
6	Clusters (Speech Acts) (ours)	76.0	76.2	59.2	59.8
7	Clusters (Speech Acts) + Politeness Strategies	78.4	77.2	62.8	64.4
8	Clusters (Speech Acts + words)	80.5	82.2	68.4	66.7
9	Clusters (Speech Acts + words) + unigrams	82.5	82.7	69.5	71.3
10	Clusters (Speech Acts + words) + unigrams + Politeness Strategies	81.3	82.9	69.4	69.4
11	Fine-tuned BERT		**89.1**		**75.5**

Table 3: Accuracy on the test set for each model and the features used to produce the results in both the in-domain and cross-domain settings. The highest accuracies among explainable models are underlined, while the highest overall accuracies are in bold.

performance but not cross-domain in comparison with Politeness Strategies (Row #2). Using speech act clusters alone (Row#6) is insufficient and produces a slightly lower accuracy than the Politeness Strategies (Row#1) in both settings. Combining speech acts clusters with unigrams and word clusters (Row#9) results in the highest accuracy overall in the cross-domain setting (71.3% for LR) among explainable models. This represents a 4.1% gain in accuracy compared with the best baseline model (Row#2 - SVM). However, speech act features fall short in the in-domain setting compared to word clusters. A fine-tuned BERT (Devlin et al., 2018) still performs best for this task in both domains. However, it is not considered explainable. We notice LR performs generally better than SVMs when combining different feature sets (Rows #5, #9 and #10). This could be due to its probabilistic nature, which helps in handling combining different feature sets effectively, especially with sparse features. In other cases, we did not observe a pattern on which model performs better. Overall, speech acts help improve cross-domain performance, but those features alone aren't sufficient and need to be combined with word-level features (e.g. word clusters and unigrams). While there is still a gap in performance between explainable models and more complex ones, our results highlight that knowledge from LLMs can improve explainable models.

5 Conclusion

We introduced a method for improving the generalizability of explainable politeness detection models. The method discovers interpretable features based on speech acts assigned by GPT-4. Those features improve over existing methods in the cross-domain setting while falling short in the in-domain setting. This work shows that LLMs can enhance explainable models.

6 Limitations

One of the limitations of our method is that the speech act categories must be pre-determined by the practitioner. The need and significance of speech acts can vary depending on the task. In this work, we used existing literature on politeness to determine the appropriate speech acts. Moreover, speech acts are open to interpretation. Their interpretation depends on different factors, including the context of the conversation. This makes evaluating the performance of GPT-4 in labeling speech acts challenging. For example, the phrase *What's the problem?* could be interpreted as a direct question in one context, but as an expression of a negative sentiment in another context. Finally, LLMs outputs can be inconsistent between different runs, which may impact the reproducibility of the results obtained in this work. The politeness annotators in the Wikipedia and Stack Exchange datasets were all U.S.-based and all data is in English. Since politeness can vary culturally, there is a risk that the proposed model may exhibit biases mainly when applied in different cultural contexts.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 Technical Report. *OpenAI*.

Sajid Ali, Tamer Abuhmed, Shaker El-Sappagh, Khan Muhammad, Jose M Alonso-Moral, Roberto Confalonieri, Riccardo Guidotti, Javier Del Ser, Natalia Díaz-Rodríguez, and Francisco Herrera. 2023. Explainable Artificial Intelligence (XAI): What we Know and What is Left to Attain Trustworthy Artificial Intelligence. *Information fusion*, 99:101805.

Ahmad Aljanaideh, Eric Fosler-Lussier, and Marie-Catherine de Marneffe. 2020. Contextualized Embeddings for Enriching Linguistic Analyses on Politeness. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2181–2190.

John Langshaw Austin. 1975. *How to do Things With Words*. Harvard University Press.

Tom B Brown. 2020. Language Models are Few-shot Learners. *Advances in Neural Information Processing Systems*.

Feng Chen, Liqin Wang, Julie Hong, Jiaqi Jiang, and Li Zhou. 2024. Unmasking Bias in Artificial Intelligence: A Systematic Review of Bias Detection and Mitigation Strategies in Electronic Health Record-based Models. *Journal of the American Medical Informatics Association*, 31(5):1172–1183.

Tobias Clement, Nils Kemmerzell, Mohamed Abdelaal, and Michael Amberg. 2023. XAIR: A Systematic Metareview of Explainable AI (XAI) Aligned to the Software Development Process. *Machine Learning and Knowledge Extraction*, 5(1):78–108.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A Computational Approach to Politeness with Application to Social Factors. *In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. 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*.

Vikas Hassija, Vinay Chamola, Atmesh Mahapatra, Abhinandan Singal, Divyansh Goel, Kaizhu Huang, Simone Scardapane, Indro Spinelli, Mufti Mahmud, and Amir Hussain. 2024. Interpreting Black-Box Models: A Review on Explainable Artificial Intelligence. *Cognitive Computation*, 16(1):45–74.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced BERT with Disentangled Attention. *arXiv preprint arXiv:2006.03654*.

Yeo Jin Kim, Halim Acosta, Wookhee Min, Jonathan Rowe, Bradford Mott, Snigdha Chaturvedi, and James Lester. 2024. Dual process masking for dialogue act recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15270–15283.

Ruizhe Li, Chenghua Lin, Matthew Collinson, Xiao Li, and Guanyi Chen. 2018. A dual-attention hierarchical recurrent neural network for dialogue act classification. *Conference on Computational Natural Language Learning (CoNLL*.

Jionghao Lin, Wei Tan, Lan Du, Wray Buntine, David Lang, Dragan Gašević, and Guanliang Chen. 2023. Enhancing educational dialogue act classification with discourse context and sample informativeness. *IEEE Transactions on Learning Technologies*.

Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. 2020. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1):18.

Yinhan Liu. 2019. Roberta: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine Learning in Python. *the Journal of machine Learning research*, 12:2825–2830.

Alec Radford. 2018. Improving Language Understanding by Generative Pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Vipul Raheja and Joel Tetreault. 2019. Dialogue Act Classification with Context-aware Self-attention. *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)*.

Eugénio Ribeiro, Ricardo Ribeiro, and David Martins de Matos. 2019. Deep dialog act recognition using multiple token, segment, and context information representations. *Journal of Artificial Intelligence Research*, 66:861–899.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339–373.

A Matching Politeness Strategies to Speech Acts

Table 5 shows how we inferred the speech acts used in this work from Danescu-Niculescu-Mizil et al. (2013)'s Politeness Strategies. Gratitude, apologizing, greeting, deference, direct start, direct question, and hedging were inferred directly from a politeness strategy of the same name. Expressing positive sentiment and expressing negative sentiment were inferred from the positive and negative lexicon features, respectively. Multiple features were used to infer the *request* speech act. Those include please start, please middle, counterfactual modals and indicative modals. This is due to those features often occurring in requests. Pronoun-based features (e.g. 1st person pronoun start) were not assigned to a speech act category since they by themselves generally do not encompass a speech act category, but are often parts of other speech acts (e.g. expressing positive sentiment such as *I like this idea*).

B GPT-4 Prompt

Table 4 shows the prompt used to extract words, phrases, and sentences associated with each speech act category. We included several instructions in the prompt to ensure consistency in the output.

C Manual Evaluation of Speech Act Labeling

We performed a manual evaluation to assess GPT-4's ability to assign speech acts. The evaluation was performed on over 100 textual elements from the Wikipedia request training set. We manually annotated those elements by assigning a speech act category to each. We then compared the annotations to the speech acts assigned by GPT-4. Figure 1 shows the confusion matrix for this evaluation. The most common error made by GPT-4 was over-predicting the hedging category (expressing uncertainty). This involved assigning textual elements to the hedging category when they were better suited as direct questions (e.g. *How come you're taking on a third language?*), expressions of negative sentiment (e.g. *Noticed you've been holding the fort alone... again*) or do not fit any (e.g. *as I have said before*).

D Experimental Details

For the classifiers, we used the scikit-learn implementation of Logistic Regression and SVM (Pedregosa et al., 2011). The Logistic Regression hyper-parameters (best are in bold) include regularization strengths of 1, 10, **100**, 1000, and 10000, regularization functions of l1 and **l2**, solvers of **liblinear**, and LBFGS. The SVM hyper-parameters (best are in bold) include regularization strengths of 1, **10**, 100 and 1000, and linear kernel functions. We don't consider other kernels like polynomials and RBF since coefficients aren't available for those. We chose the parameters that led to the highest accuracy on the development Wikipedia set. We used those to train the model and obtain the accuracy on the Wikipedia (in-domain) and Stack Exchange (cross-domain) test sets. The number of

Extract words, phrases, or sentences from the following text based on the categories listed below. For each extracted item, Place it on a separate line. Write the extracted word/phrase/sentence, followed by a colon ':', and then the category name exactly as listed below. Ensure the categories are in the same order they appear in the text. Categories: 1. Gratitude 2. Apologizing 3. Greeting 4. Expressing a positive sentiment 5. Expressing a negative sentiment 6. Hedging (expressing uncertainty) 7. Deference 8. Direct questions 9. Direct start 10. Request Output Format: Each line should be formatted as: ;Extracted text: Category'. Important Notes: Follow the category names exactly as listed above. Do not include anything beyond the extracted text and its category. If a phrase or sentence contains multiple categories, list on separate lines as above. A phrase is any meaningful sequence of words standing together as a conceptual unit. Input Text: [input text]

Table 4: Prompt for assigning speech act categories by GPT-4.

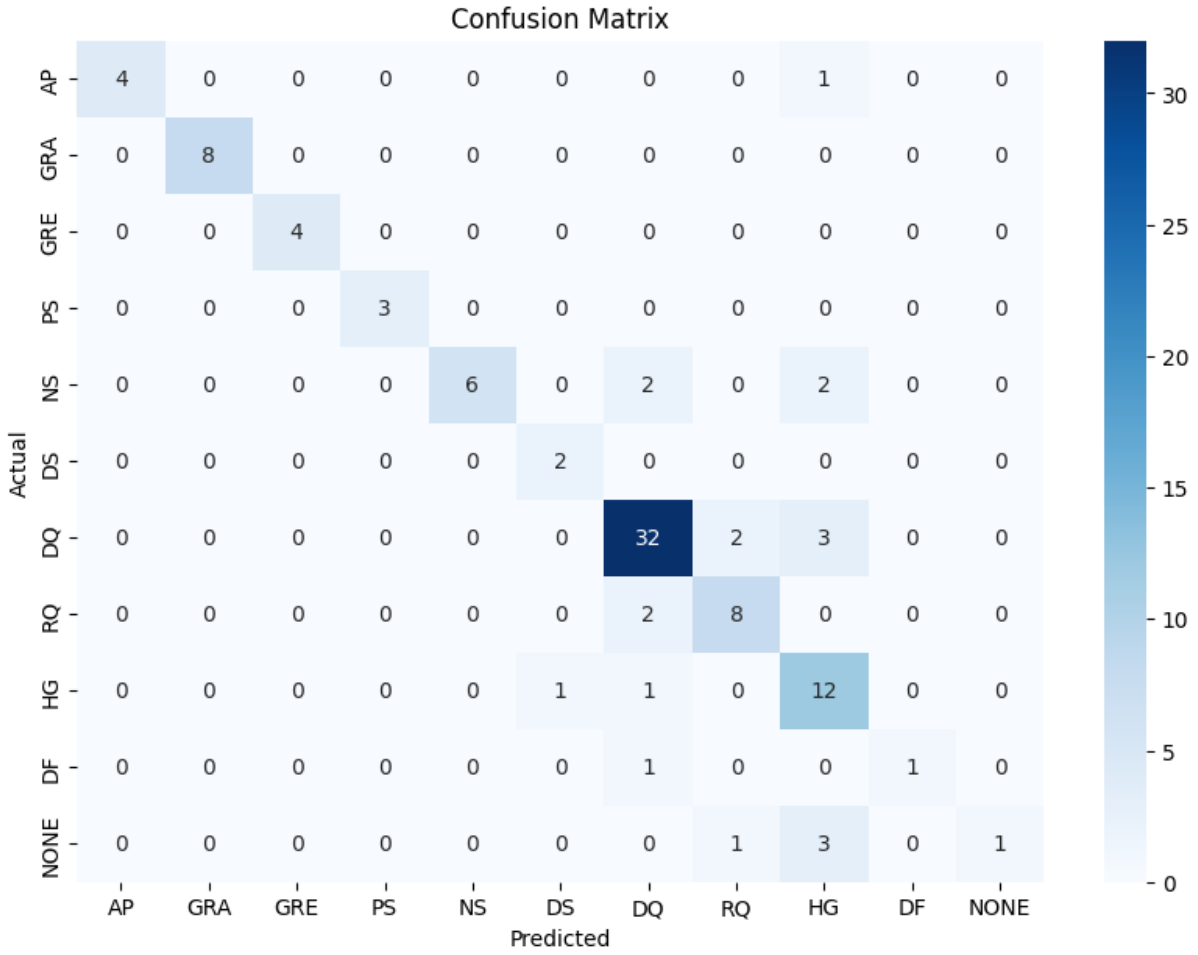

Figure 1: Confusion Matrix of a sample of speech act categories predicted by GPT-4. We used the following abbreviations for readability: AP: apologizing, GRA: gratitude, GRE: greeting, PS: positive sentiment, NS: negative sentiment, DS: direct start, DQ:direct question, RQ: request, HG: hedging, DF: deference, NONE: phrase which do not belong to any category.)

Feature	Speech Act	Example
Gratitude	Gratitude	Thank you
Apologizing	Apologizing	I am sorry
Greeting	Greeting	Hey, ...
Positive Lexicon	Expressing a positive sentiment	This is amazing
Negative Lexicon	Expressing a negative sentiment	Terrible edit
Hedges	Hedging	I suggest we move ...
Deference	Deference	Nice work!
Counterfactual models (Would/Could you)	Request	Could/Would you edit...
Indicative modal (e.g. Will/Can you ..)	Request	Can/Will you please change...
Please middle	Request	Can you please ...
Please start	Request	Please stop that...
Direct question	Direct question	What does that mean?...
Direct start	Direct Start	So that's why....
1st person pronoun	N/A	For me, it should...
1st person pronoun start	N/A	I think that ...
1st person pronoun plural	N/A	We can do....
2nd person pronoun	N/A	Can you ...
2nd person pronoun start	N/A	You just added ...
BTW	N/A	BTW I have just...
Use of *In fact*	N/A	In fact, the edits I added are...

Table 5: Politeness Strategies and inferred speech acts.

discovered speech act features (model parameters) is 43.

E More Cluster Analysis

Table 6 shows additional clusters discovered automatically with the proposed method.

Both Gratitude_1 and_2 contains polite primarily requests. However, in Gratitude_1 the speaker specifies what they are grateful for (e.g., *Thanks for your help...*), while that is not the case in Gratitude_2 (e.g., *Thanks - can you ...*).

Direct start_1 represents a cluster of element embeddings from impolite requests containing a direct start to the request or a sentence (e.g., *Is it your view..*). Direct start_2 contains a cluster of element embeddings from requests that lean impolite and include a direct start (e.g., *So you argue that as long as you follow*).

Apologizing_1 represents a cluster of element embeddings from polite requests that contain apologetic expressions (e.g., *apologies*) where the speaker expresses in the request what they are apologizing for (e.g.*Sorry if it seems like I'm bugging you ...*). Apologizing_2 also contains element embeddings from a few primarily impolite requests, which a form of apology (*e I should not have used rollback*) followed by a confrontational statement *I was aware of the problem, but you had not solved it, had you?*.

F Use of AI Assistant

ChatGPT was used to refine the language of the paper and discover potential bugs in code.

Cluster ID	Items	Polite %	Example	Label
Gratitude_1	80	100%	**Thanks** for <url>. Do you think you could do the same for <url> and <url>?	+
			Thanks for your help on <url>. Any advice on edits we should make to get it up the quality scale?	+
Gratitude_2	33	82%	**Thanks**. Beer?	+
			Thanks–can you add the romanization of the Arabic at <url> too? Is it "jibna baladi"?	+
Direct Start_1	14	0%	Well, that's probably going to be an annoying AfD. **How about <url> and <url>?**	-
			Is it your view that the United States Senate doesn't have "any credibility"? Or are you of the view that some cyber truck that zooms around the WWW must have bumped into the web page by "accident" and dumped stuff endorsed the United States government?	-
Direct Start_2	42	43%	**So your argument is that as long as you follow** the "rules," nothing else should matter? That if it's legal it's OK?	-
			Thank you, I was wondering what type of unit this was. **Civil Defence**, some kind of police squad, or perhaps even Freikorps?	+
Apologizing_1	17	76%	**Sorry** if it seems like I'm bugging you about the above image - I'm trying to determine the copyright holder, which needs to be specified on the image page per <url>a. When I click on the source link, I get an "access forbidden" message. Can you please specify the copyright holder on the page?	+
			I am sorry, got confused on who created this article. :-/ **Apologies**?	+
Apologizing_2	4	25%	I suppose I **should not have used rollback**. I was aware of the problem, but you had not solved it, had you?	-
			I obviously made a mistake too. Do you really think I'd purposefully misinterpret things like that?	-

Table 6: Example clusters discovered with our method. The columns show: the cluster ID, number of items, percentage of polite items, example elements, and labels (+ for polite, - for impolite). Bold text highlights the key element in the example.