When to generate hedges in peer-tutoring interactions

Alafate Abulimiti[1,2], **Chloé Clavel**[3], **Justine Cassell**[1,4]

[1] INRIA, Paris [2] ENS/PSL <alafate.abulimiti@inria.fr>
[3] LTCI, Institut Polytechnique de Paris, Telecom Paris <chloe.clavel@telecom-paris.fr>
[4] Carnegie Mellon University <justine@cs.cmu.edu>

Abstract

This paper explores the application of machine learning techniques to predict where hedging occurs in peer-tutoring interactions. The study uses a naturalistic face-to-face dataset annotated for natural language turns, conversational strategies, tutoring strategies, and nonverbal behaviors. These elements are processed into a vector representation of the previous turns, which serves as input to several machine learning models, including MLP and LSTM. The results show that embedding layers, capturing the semantic information of the previous turns, significantly improves the model's performance. Additionally, the study provides insights into the importance of various features, such as interpersonal rapport and nonverbal behaviors, in predicting hedges by using Shapley values (Hart, 1989) for feature explanation. We discover that the eye gaze of both the tutor and the tutee has a significant impact on hedge prediction. We further validate this observation through a follow-up ablation study.

1 Introduction

Effective communication involves various conversational strategies that help speakers convey their intended meaning and manage social interactions at the same time. These strategies can include the use of self-disclosure, praise, reference to shared experience, etc. (Zhao et al., 2014). Hedges are one of those strategies that is commonly used in dialogue. Hedges are words or phrases that convey a degree of uncertainty or vagueness, allowing speakers to soften the impact of their statements and convey humility or modesty, or avoid face threat. Although hedges can be effective in certain situations, understanding when and how to use hedges is essential and challenging.

The use of hedges is especially significant in tutoring interactions where they may facilitate correcting a wrong answer without embarrassing the recipient. However, the use of hedges in this context is not limited to expert educators. They are also found to be abundant in peer-tutoring settings. In fact, Madaio et al. (2017a) found that confident tutors tend to use more hedges when their rapport with the tutee is low, and that this pattern leads to tutees attempting more problems and solving more problems correctly. Hence, the detection and correct deployment of hedges, at the right time, is not just pleasant, but crucial for the development of effective intelligent peer tutoring systems.

While the use of hedges in conversation is an important aspect of effective communication, automatically generating hedges in real-time at the right time, can be a challenging task. In recent years, there have been several studies of automatic hedge detection (Raphalen et al., 2022; Goel et al., 2019), particularly in the context of dialogue systems. However, despite significant advances in detection, generating hedges in a timely and appropriate manner remained unsolved. For example, the RLHF-based training method enables the development of robust language models that align with human preferences (Ouyang et al., 2022). However, this approach does not explicitly instruct large language models (e.g., ChatGPT) in pragmatic and social skills, such as the appropriate use of hedges during communication. This lack of specific training can result in a gap in the model's ability to effectively integrate these conversational nuances into its responses in *at the correct time*. This limitation can affect the quality of communication and highlights the need for further research on effective hedge strategie generation; that is, to generate hedges at the right time.

Despite the widespread use of hedges in communication, there is still much to learn about their timing and the effectiveness of their use, particularly in dialogue rather than running text, and specifically in the current article, in peer-tutoring environments.

To address this gap in the literature, our research

Proceedings of the 24th Meeting of the Special Interest Group on Discourse and Dialogue, pages 572–583
September 11–15, 2023. ©2023 Association for Computational Linguistics

focuses on two key questions:

RQ1: First, can we predict when hedges should be generated in peer-tutoring environments?

To address this question we investigate whether it is possible to identify the points at which hedges should be introduced during a peer tutoring dialogue.

RQ2: Second, what features contribute to accurate predictions? of where to place hedges?

To address this question we focus on the explainability of classification models using Shapley values (Sundararajan and Najmi, 2020) .

2 Related Work

2.1 Hedges

Hedges are a common rhetorical device used to diminish the impact of an utterance, often to avoid unnecessary embarrassment on the part of the listener or to avoid the speaker being interpreted as rude. In linguistic terms, hedges diminish the full semantic value of an expression (Fraser, 2010). **Propositional hedges**, also called *Approximators*, refers to the use of uncertainty (Vincze, 2014), vagueness (Williamson, 2002), or fuzzy language (Lakoff, 1975), such as "sort of" or "approximately". On the other hand, **Relational hedges** are used to convey the subjective or opinionated nature of a statement, such as "*I guess* it will be raining tomorrow". **Apologizer** (Raphalen et al., 2022; Goel et al., 2019; Fraser, 2010) is an expression used to mitigate the strength of an utterance by using apologies, is another type of hedges. such as "*I am sorry*, but you shouldn't do that." Although the different types of hedges function differently, they all share a common role of mitigation in conversation. Therefore, in this paper, we focus on simply predicting hedges vs non-hedges.

As described above, in tutoring, including peer tutoring, hedges are frequently used and have a positive impact on performance (Madaio et al., 2017a). Powerful language models such as GPT-4 (OpenAI, 2023) and ChatGPT (OpenAI, 2022) are now capable of generating hedges with appropriate prompts, but these language models do not actively generate hedges (Abulimiti et al., 2023), fIn other words, the question of how to use thedges correctly in the next conversational action remains unsolved.

2.2 Conversational Strategy Prediction

The development of approaches for predicting conversational strategies – or particular ways of saying things – has progressed significantly over the past few years in the field of dialogue systems. Early studies, such as the COBBER, a domain-independent framework, used a Conversational Case-Based Reasoning (CCBR) framework based on reusable ontologies (Gómez-Gauchía et al., 2006). The aim was to help people use a computer more effectively by keeping them in the right mood or frame of mind. Methods such as reinforcement learning have also been introduced in non-task-oriented dialog systems, including a technique known as policy learning (Yu et al., 2016). Reinforcement learning has been explored, as well,for training socially interactive agents that maximize user engagement (Galland et al., 2022).

The Sentiment Look-ahead method is used to predict users' future emotional states and to reward generative models that enhance user sentiment (Shin et al., 2020). The rewards include response relevance, fluency, and emotion matching. These rewards are built using a reinforcement learning framework, where the model learns to predict the user's future emotional state. Romero et al. (2017) designed a social reasoner that can manage the rapport between user and system by reasoning and applying different conversational strategies.

More recently, deep learning-based approaches have emerged. For example, the Estimation-Action-Reflection (EAR) framework combines conversational and recommender approaches by learning a dialogue policy based on user preferences and conversation history (Lei et al., 2020).

Perhaps the most recent advances in the field have focused on how to create an empathetic dialogue system. MIME (Majumder et al., 2020) used the emotion mimicry strategy to match the user's emotion based on the text context. EmpDG (Li et al., 2020) generated empathetic responses using an interactive adversarial learning method to identify whether the responses evoke emotional perceptivity (the ability to perceive, understand, and be sensitive to the emotions of others.) in dialogue. The Mixture of Empathetic Listeners (MoEL) model (Lin et al., 2019) generates empathetic responses by recognizing the user's emotional state, using emotion-specific multi-agent listeners to respond, and then combining these responses based on the emotion distribution. This process effectively merges the output states of the listeners to create an appropriate empathetic response. The model then crafts an empathetic response grounded in the user's emotions, which are monitored by the emotion tracker. Despite the notable success of MIME and MoEL in predicting emotions or conversational strategies, they do not incorporate the social context (e.g., the relationship between speakers), or the emotional tenor of the conversation up until that point, nor do they include important nonverbal behaviors into reasoning and decision-making processes. However, such elements are fundamental for the correct use of social language, and their absence potentially limits the effectiveness and naturalness of these models.

Predicting the appropriate emotion or conversational strategies in a conversation is a challenging task, mainly because determining what is "appropriate" in a conversation is rather subjective and is certainly context-dependent. For example, EmpDG (Li et al., 2020) model achieved an accuracy of approximately 0.34 across the 32 evenly distributed labels in the Empathetic Dialogue dataset (Rashkin et al., 2019). indicating the complexity of the problem at hand. Similarly, MoEL (Lin et al., 2019) model achieved varying degrees of accuracy in the same dataset - 38% for the top 1, 63% for the top 3, and 74% for the top 5 for emotion detection, further emphasizing the difficulty of the task.

The current paper aims to fill the lacunae in prior work by integrating social context and nonverbal behaviors as predictive features to construct predictive models for hedges.

3 Methodology

3.1 Task Description

Suppose we have a set of dialogues $D = \{d_1, d_2, d_3, ...d_n\}$. Each dialogue $d = \{u_1, u_2, u_3...u_m\}$ consists of m turns, with u_i representing a specific turn. Both tutor and tutee turns in these dialogues can be categorized as either hedges or non-hedges. However, for the purposes of our analysis, we will primarily focus on the tutor's turns. The label of a particular turn u_i is denoted as l_i. Furthermore, every turn can be depicted as a feature vector X, composed of elements $(x_1, x_2, ..., x_N)$. Here, N signifies the total number of features used to characterize a turn. Each turn in the dialogue is assigned a fixed window size (ω) of the dialogue history, represented as: $h_i = \{u_{max(1,i-\omega)}, u_{i-\omega+1}, ...u_i\}$. The primary objective of this research is to develop a model, denoted M, capable of predicting the type of hedge l'_{i+1} that a tutor will use next, based on the dialogue history h_i. The effectiveness of the model is measured using standard classification metrics, such as precision, recall, and F1 score.

Predicting hedges in a peer-tutoring conversation can be simplified to a binary classification problem. The features used as inputs are extracted from the turns in the interaction (further details in Section 3.3), while the output is a binary value showing whether or not hedges are present in each turn.

3.2 Corpus

The dataset used in the current work is the same as that employed in our previous work on hedges (Madaio et al., 2018). It is a subset of a larger investigation into the role of social, rapport-building conversational strategies in task-oriented dialogue. The corpus consists of face-to-face interaction from 20 same-gender dyads of American teenagers, with an average age of 14.3 years (and a range of ages from 13 to 16 years), gender-balanced [1] , and recorded twice over two weeks. However, due to technical issues, data from only 14 dyads' data were usable. The participants were asked to to take turns tutoring one another in different aspects of linear algebra. Each hour-long session was divided into 4 phases: an initial social period, followed by a first peer tutoring period, then a second short social period, and finally, the teens switched roles, with the tutee becoming tutor for the second task period. For the 14 dyads we used for our model, 28-hour-long face-to-face interactions were recorded over the period of two weeks. The recorded video and audio data were transcribed, resulting in approximately 9479 turns for the 14 dyads. These included 8399 non-hedges and 1080 hedges. 4214 non-hedges and 507 hedges in the tutors' turns since, as described above, we looked only at tutor hedges for this analysis (although note that both tutor and tutee hedges in prior turns were used as input). A "hedge turn" is any turn that includes hedging language. We also retained non-speech segments such as laughter and fillers.

Peer tutoring is a popular teaching method used in many schools and educational settings. As described above, and in Madaio et al. (2017b), even

[1] The corpus used here comes from earlier work by the last author and her colleagues, as cited above, and was used in accordance with the original experimenters' Institutional Review Board (IRB) approval. That approval required that the children's data not be released, which means that we cannot share the corpus. However, a pixelated example of the video data is available at github.com/neuromaancer/hedge_prediction.

though these teenagers may be inexperienced, in contexts of low rapport, when they use hedges during tutoring, their tutees are encouraged to attempt more problems and succeed in solving more of them. This positive outcome justifies the use of this dataset for studying hedges in tutoring interactions. While we recognize the importance of exploring the use of hedge with expert tutors in the future, our current focus on untrained peer tutors provides a unique perspective on how hedges can impact learning, even when the tutors themselves are not highly experienced. The methods and results from our study can be used as a foundation for future research, which could include the investigation of expert tutors and the potential differences in their use of hedges.

3.3 Features

In this section, we outline the features used as input vectors (i.e., u_i vector) for our prediction model, which seeks to properly predict the hedging strategy for the tutor's upcoming turn. In total, we have a vector with a length of 438 to represent a turn.

3.3.1 Turn embedding

Turn embedding is a common technique in natural language processing that involves representing a turn as a vector. In this study, we apply a sentence transformer (Reimers and Gurevych, 2019) to generate turn embeddings from the tutor-tutee conversation. This feature enables us to capture the semantic meaning of the turn in the context of the conversation, which can be helpful for predicting hedges.

3.3.2 Conversational Strategies (*CS*) of the previous turns

Conversational strategies refer to the different ways of speaking used by both speakers to manage social interaction. Strategies considered in this study are self-disclosure, praise, violation of social norms, and hedges. Self-disclosure (Derlega et al., 1993) refers to situations in which the tutor or tutee shares personal information, which is often used to build rapport. Praise (Brophy, 1981) is a form of positive feedback that acknowledges and reinforces the other person's behaviors or attributes. Violation of social norms (Zhao et al., 2014), which in this population often consists of friendly teasing, is a conversational move in speaker demonstrates the special nature of the relationship with the listener by engaging in slightly transgressive behavior. The conversational strategy annotation was carried out by Madaio et al. (2018), and inter-rater reliability achieved a minimum Krippendorff's alpha of over .7 for all strategies.

In terms of hedges, we note that we only use the speakers' previous hedge strategies to predict the tutor's next hedge strategy. This avoids any issues with predicting label leakage.

3.3.3 Tutoring Strategies (*TS*) in the previous turns

Tutoring strategies (Madaio et al., 2016) refer to the different techniques employed by the tutor or tutee to facilitate learning. Strategies considered in this study include deep/shallow questions, meta-communication, knowledge building, and knowledge telling. The deep question encourages critical thinking and higher-order cognition. The shallow question is used to confirm or clarify understanding. Meta-communication is a strategy whereby the tutor or tutee refers to the tutoring process or the tutor/tutee's self-evaluation of their own knowledge, which can help to clarify misunderstandings and promote effective communication. Knowledge building involves introducing new concepts or ideas, discussing the reasoning-mathematical solving steps, and providing examples. Knowledge telling refers to providing information (i.e., simply stating numbers, variables). The tutoring strategies annotation was also carried out by Madaio et al. (2018), with annotators achieving a minimum Krippendorff's alpha of .7 for all tutoring strategies.

3.3.4 Dialogue Act (*DialAct*) of the previous turns

Dialogue acts are types of speech acts (Searle, 1965) used by tutors and tutees during their interactions. In our study, we use the widely-used DAMSL (Dialogue Act Markup in Several Layers) (Jurafsky, 1997) coding schema to annotate dialogue turns by using a state-of-the-art dialogue act classifier with context-aware self-attention (Raheja and Tetreault, 2019). In our dataset, only 6 dialogue acts were found, they are *Abandoned* or Turn-Exit (%) , *Acknowledge (Backchannel)* (*b*), *Backchannel in question form* (*bh*), *Yes-No-Question* (*qy*), *Statement-non-opinion* (*sv*) and *Statement-opinion* (*sd*).

3.3.5 Rapport in the previous turns

As our previous work demonstrates, the level of rapport between tutor and tutee plays a role in the use of hedges. We therefore include it as a feature in our study. Rapport is “The relative harmony of relations felt by both participants” (Spencer-Oatey, 2005). The rapport annotation was carried out by Amazon Mechanical Turk (AMT) annotators as described in Madaio et al. (2018). Rapport level was operationalized as a 7 point Likert scale, where a higher score indicates a stronger level of rapport. For the annotation of rapport, the annotators employed the “thin slice” method (Ambady and Rosenthal, 1993), whereby the experimenter segmented each video into 30-second clips and randomized the order. To ensure the quality of rapport annotations, three annators evaluated each clip, and the experimenter applied the inverse-bias correction method (Parde and Nielsen, 2017) for selecting a single score for each clip. In the current study, when the dialogue history is contained within a single slice, we directly use the annotated rapport level of that particular slice as the historical rapport level. However, if the dialogue history extends over two slices, we select the rapport level of the slice containing the majority of the dialogue history.

3.3.6 Nonverbal Behaviors (NB)

Nonverbal behaviors, such as head nod, smile, and gaze, are an essential aspect of interpersonal communication that can also contribute to the development of rapport (Tickle-Degnen and Rosenthal, 1990). The gaze and smile annotation was carried out by Madaio et al. (2018), we annotated the head nods with 2 annotators. All the annotations were carried out after annotators reached an inter-rater reliability of 0.7 or above on Krippendorff’s alpha. We collected all nonverbal behaviors that occurred during one turn and encoded them using one-hot encoding. For head nods and smiles, we used a binary labeling approach, marking 1 for their occurrence and 0 for non-occurrence. As gaze serves as a potent indicator of attention, we categorized it into 4 distinct types: no gaze appeared in the video, gaze at partner, gaze at worksheet, and gaze elsewhere.

Mutual gaze between interlocutors, mutual smiles, and mutual head nods serve as great indicators of alignment and rapport in communication. These are not encoded separately, as our encoding process for nonverbal behaviors captures the behaviors of both participants within a turn, not only the current turn holder. Our current approach successfully captures these important mutual signals.

3.3.7 Contextual Information (*ConInfo*) in the previous turns

Our model also incorporates contextual information that characterizes the discourse environment between the two interlocutors. Specifically, we include features such as the session and period numbers, which help to encapsulate the temporal dynamics of the tutoring interactions. We also consider the math problem ID and the correctness of the current problem response, which act as markers of the present learning context. These features can illuminate the complexity of the ongoing problem and the students’ performance, potentially influencing their use of hedges. The tutee’s and tutor’s pre-experiment test scores are also included, serving as initial measures of their knowledge before the tutoring session. This data can help to identify the starting knowledge disparity between the tutor and the tutee. It is plausible that these pre-test scores might also be linked with the students’ level of confidence, which could subsequently impact their use of hedges (Madaio et al., 2017a).

Norman et al. (2022) suggested a link between verbal alignment signals, such as backchannels (e.g., “um”, “hhm”, “oh..”), and learning gains in a cooperative learning environment. Given the role of hedging as a social language skill that improves learning performance, we hypothesize its connection to dynamic learning gains. Consequently, we incorporated the frequency of these verbal alignment signals from the previous four conversational turns into our model input.

3.4 Vector Representation

Before presenting the specific models, we first describe how we convert each sequence of turns into a vector representation. Our vector representation consists of three basic parts: turns as a sequence of tokens, annotations based on the turn (e.g., conversational strategies), and the nonverbal behaviors. Figure 1 shows that we divide a vector of turns into 6 parts: turn embedding, conversational strategies (*CS*), tutoring strategies (*TS*), nonverbal behaviors (*NB*), contextual information (*ConInfo*) and dialogue acts (*DialAct*). After encoding each turn in this fashion, we use the four previous turns as a history tensor of a turn. This history ten tensor will be the input to the prediction models, and the model’s output will be this turn’s hedge label.

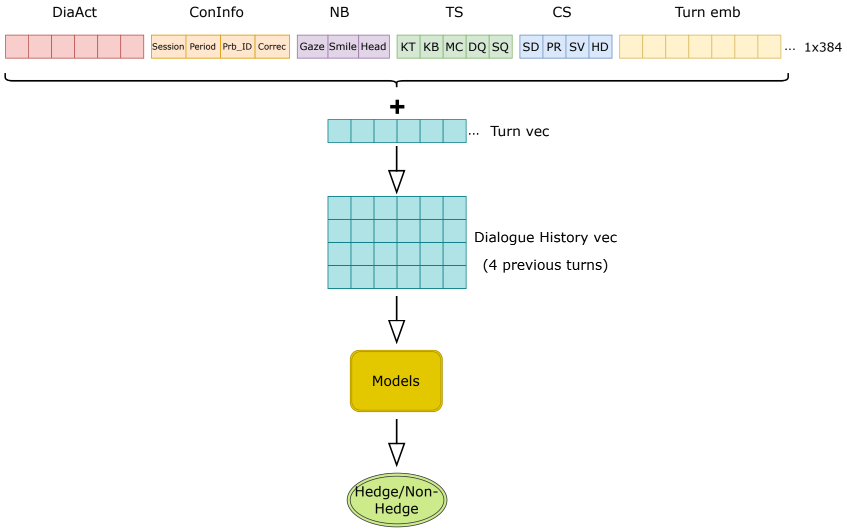


Figure 1: Vector Representation

3.5 Prediction as Classification

We mentioned in the previous section that we transform the prediction problem into a classification problem. This means that the corresponding hedge strategy is obtained by classifying different previous interactions (i.e., dialogue history) and historical characteristics (e.g., rapport, etc.). The classification models used are presented here.

The selection of learning models in this study is strategic and based on our research objectives. Our primary aim is not to engineer a perfect system for hedging. Instead, we seek to comprehend the variables that influence hedging in dialogue. As such, our approach leans towards the use of models that are effective in contextual understanding. For example, Long Short-Term Memory networks (LSTMs) were chosen over Multi-Layer Perceptrons (MLPs) due to their superior ability to manage and interpret context, an essential factor in our exploration of hedging phenomena.

3.5.1 LightGBM

In this work, we used LightGBM (Ke et al., 2017), a gradient boosting framework known for its efficiency. We use it to predict hedges in dialogues, relying only on dialogue features such as conversational strategies, tutoring strategies, nonverbal behaviors, and contextual information, while turn embeddings are not included.

3.5.2 XGBoost

We also used the Extreme Gradient Boosting (XGBoost) algorithm (Chen and Guestrin, 2016), which is a decision tree-based ensemble machine learning algorithm that uses a gradient boosting framework. Similar to LightGBM, the turn embedding is not used.

3.5.3 Multi-layer perceptron (MLP)

We constructed a multi-layer perceptron using two sets of features. These included a pre-trained contextual representation of the turn, specifically from the SentBERT model (Reimers and Gurevych, 2019) which is the most prevalent sentence embedding tool, and the concatenation of all the features mentioned in Section 3.3.

3.5.4 Long Short-Term Memory (LSTM)

We use the same features and apply them to LSTM (Hochreiter and Schmidhuber, 1997) and also LSTM with attention (Bahdanau et al., 2015). LSTM has a good ability to capture temporal correlations, and we expect this ability to enhance prediction performance.

3.6 Implementation Details

In order to address the imbalance in our dataset, where the ratio of hedge to non-hedge instances is approximately 1:10, we used the Synthetic Minority Over-sampling Technique (SMOTE) (Chawla et al., 2002) for each model to augment our learning process. SMOTE is a popular method that generates synthetic examples in a dataset to counteract its imbalance. Given the variable nature of model performance, we implemented a 5-fold cross-validation strategy to evaluate the models. In order to account for the imbalanced nature of the dataset, we opted to use a lower number of folds in

Models	F1-score	Precision	Recall
LightGBM (w/o emb)	0.24 (±0.07)	0.17 (±0.03)	0.45 (±0.07)
XGBoost (w/o emb)	0.24 (±0.07)	0.16 (±0.03)	0.45 (±0.07)
MLP	0.25 (±0.06)	0.16 (±0.03)	0.52 (±0.07)
MLP (only emb)	0.26 (±0.05)	0.16 (±0.02)	0.74 (±0.06)
MLP (w/o emb)	0.26 (±0.06)	0.17 (±0.06)	0.56 (±0.07)
LSTM	0.25 (±0.06)	0.16 (±0.03)	0.50 (±0.07)
LSTM (only emb)	0.28 (±0.07)	0.19 (±0.08)	0.52 (±0.07)
LSTM (w/o emb)	0.25 (±0.05)	0.15 (±0.02)	0.75 (±0.06)
AttnLSTM	0.24 (±0.06)	0.15 (±0.03)	0.57 (±0.07)
AttnLSTM (only emb)	0.25 (±0.07)	0.17 (±0.03)	0.45 (±0.07)
AttnLSTM (w/o emb)	0.23 (±0.06)	0.15 (±0.07)	0.57 (±0.07)
Dummy	0.11 (±0.08)	0.14 (±0.06)	0.10 (±0.04)

Table 1: Comparison of MLP and LSTM models for predicting hedges

the cross-validation process. By choosing 5 folds instead of a higher number, we aimed to ensure that each fold would contain a sufficient representation of samples from each class. The model that delivered the best performance during this cross-validation process was then chosen to make predictions on the test set. For the neural models, we adjusted the loss function to account for class imbalance, thereby compelling the models to accommodate less frequent classes more effectively. The code is available in https://github.com/neuromaancer/hedge_prediction

4 Results

4.1 Classification Results

To answer the research question 1, we conducted classification experiments on different models. Table 1 offers an in-depth comparison of multiple machine learning models for predicting hedges in a peer-tutoring dataset. We also incorporated a dummy classifier for comparison, which generates predictions in accordance with the class distribution observed in the training set. The performance metrics are F1 score, precision and recall, all of which include confidence intervals ($\alpha = 0.05$). The dataset is composed of several types of input features described in Section 3.3. The models used different combinations of these inputs. (w/o emb) indicates that the model uses only the features without turn embeddings. If not specified, the model uses all features plus turn embeddings.

From Table 1, the LightGBM and XGBoost models without embeddings achieved relatively low scores for F1 scores, precision and recall, indicating limited performance in terms of balanced precision and recall. The MLP models, particularly those using only embeddings, showed a remarkable recall of 74%, but at the cost of reduced precision. The LSTM model using only turn embeddings demonstrated balanced performance across all metrics, achieving the highest precision of 19% and a competitive F1 score of 0.28. However, the attention-based LSTM (AttnLSTM) model did not significantly outperform the standard LSTM model in any metric.

The inclusion of turn embeddings significantly impacts model performance. Models with only embeddings perform better in terms of F1 score and recall, suggesting that the semantic information captured in these embeddings, which represented the semantic information of turns, is crucial for hedge prediction. Second, models without embeddings also performed reasonably well in F1 score, implying that other features such as rapport, conversational strategies, tutoring strategies, nonverbal behaviors, and contextual information are also important. These features should not be overlooked.

The LightGBM and XGBoost models, which only use features without turn embeddings, also display competitive performance compared to the MLP, LSTM, and AttnLSTM models using all features. This suggests that although turn embeddings provide valuable information for hedge prediction, models can still achieve satisfactory results even without them. The AttnLSTM models, which incor-

Model \ Feature	*N/A*	*Rapport*	CS	*TS*	*NB*	*ConInfo*	*DialAct*
XGBoost	0.24 (±0.07)	0.15 (±0.08)	0.10 (±0.08)	0.15 (±0.09)	**0.08** (±0.07)	0.10 (±0.08)	0.12 (±0.08)
LightGBM	0.24 (±0.07)	0.16 (±0.08)	**0.09** (±0.08)	0.10 (±0.07)	0.10 (±0.10)	0.12 (±0.09)	0.13 (±0.08)
LSTM	0.25 (±0.05)	0.24 (±0.05)	0.26 (±0.06)	0.24 (±0.06)	0.22 (±0.06)	0.25 (±0.07)	**0.21** (±0.06)
AttnLSTM	0.23 (±0.06)	**0.20** (±0.06)	0.22 (±0.05)	0.25 (±0.05)	0.24 (±0.05)	0.23 (±0.07)	0.22 (±0.06)
MLP	0.26 (±0.06)	0.25 (±0.06)	0.25 (±0.06)	0.26 (±0.06)	0.25 (±0.06)	0.27 (±0.06)	**0.21** (±0.07)

Table 2: F1 scores after the feature ablation, *CS*: Conversational Strategies; *TS*: Tutoring Strategies; *NB*: Nonverbal Behaviors; *ConInfo*: Contextual Information; *DialAct*: Dialogue Act.

porate attention mechanisms, do not show significant improvements over the regular LSTM models. This could be due to the limited amount of data available, which cannot unleash the potential of the attention mechanism.

Since good performance can also be achieved using the extracted features, in order to answer our research question 2, in the next subsections we will mainly investigate the importance of features in predicting hedges.

4.2 Features Explanation with Shapley values

Shapley values (Hart, 1989), originating from cooperative game theory, have emerged as a powerful model-agnostic tool to explain the predictions of machine learning models. This approach provides a way to fairly distribute the contribution of each feature to the overall prediction for a specific instance. By calculating the Shapley value for each feature, we gain insight into the importance of individual features within the context of a specific prediction. This interpretability technique has been adopted across various machine learning models. In this study, we use Shapley values to interpret the contributions of extracted features in our classification models using the SHAP python package (Lundberg and Lee, 2017).

Figure 2 in the Appendix illustrates the importance of each feature for prediction when only features are used as input to different prediction models. The importance of features within the models can differ depending on their architectures. For simplicity, we identify the features that frequently appear in these 4 figures as significant indicators. Therefore, we have selected some of the most representative features in predicting hedges in Table 3.

Features	**Valence**
correctness	+
no gaze from tutor	-
problem id	-
rapport	-
tutee's deep question	-
tutee's gaze at tutor	-
tutee's pre-test	-
tutor's gaze at elsewhere	-
tutor's praise	-

Table 3: Features and their Valences

Based on Table 3, certain features have a significant impact on the likelihood of using hedges in tutoring conversations. Rapport has a negative valence, suggesting that higher rapport between the participants results in a lower likelihood of hedges being used. This confirms the finding cited above, that hedges are more frequent in low rapport interaction (Madaio et al., 2017c). Interestingly, the "problem ID" feature also has a negative valence, indicating that as the complexity or difficulty of the problem increases, the likelihood of using hedges decreases. This could be because tutors tend to be more assertive or confident when addressing more challenging problems.

Moreover, certain conversational features such as "tutee's deep question" and "tutor's praise" have a negative valence, implying that these actions tend to decrease the likelihood of hedges. This could be because deeper questions or praise might indicate a more open and confident dialogue, thus reducing the need for hedges.

The table also reveals a negative correlation between various non-verbal cues such as "no gaze from tutor", "tutee's gaze at tutor", and "tutor's gaze at elsewhere", and the occurrence of hedges. When the tutor is not gazing at the tutee, the likelihood of hedges decreases. The tutee's gaze at the tutor and the tutor's gaze at elsewhere are negatively associated with the use of hedges. This could indicate that when tutors' attention is focused elsewhere, they are attending less to how best to convey instruction or correction. To our knowledge, this is the first demonstration that specific nonverbal cues substantially influence the likelihood of a hedge being used in the succeeding turn of peer-tutoring interactions.

4.3 Ablation Study

We next examine the aforementioned models with different features ablated from input. This approach allows us to identify which features, when absent, lead to the best or worst performance in each model, implying that these features may not have contributed positively (or negatively) to the model's performance. Our study considered 6 groups of features: Conversational Strategies (*CS*), Tutoring Strategies (*TS*), Nonverbal Behaviors (*NB*), Contextual Information (*ConInfo*), Dialogue Act (DialAct), and Rapport.

Table 2 shows the different F1 scores as a consequence of removing the different features. For XGBoost and LightGBM, the worst performance is observed when *NB* and *CS* were removed, respectively, which implies that these features may provide important information for these models. The LSTM and MLP models showed a significant drop in performance when the *DialAct* feature was removed, suggesting a substantial dependency of these models on the *DialAct* feature for their prediction capabilities. Interestingly, the best performance of AttnLSTM was achieved when the rapport feature was removed, suggesting that the attention mechanism could compensate for loss of rapport.

5 Conclusion and Future Work

This study presents an effective approach to predict where hedges occur in peer-tutoring interactions using classic ML models. Our results show the importance of considering various types of input features, such as turn embeddings, rapport, conversational strategies, tutoring strategies, nonverbal behaviors, and contextual information. Moreover, Shapley values applied to the predictions of the different models show, for the first time, that the gaze of both tutor and tutee may play a critical role in predicting hedges. This observation is substantiated by subsequent ablation studies, where classic classification models, like XGBoost and LightGBM, experienced a significant decline in F1 score when removing nonverbal behavior features.

For future work, several directions can be pursued. First, the investigation of hedge generation in the context of expert tutors could provide valuable insights into how experienced tutors use hedges differently and how these differences might affect learning outcomes. Second, incorporating reinforcement learning techniques to enhance specific aspects of the interaction, such as learning performance, could improve the practical applications of our findings.

Acknowledgments

We thank the anonymous reviewers for their helpful feedback. We express sincere gratitude to the members of the ArticuLabo at Inria Paris for their invaluable assistance in the successful completion of this research, and to the members of the ArticuLab at Carnegie Mellon University in Pittsburgh for answering our questions about their prior work. This study received support from the French government, administered by the Agence Nationale de la Recherche, as part of the "Investissements d'avenir" program, with reference to ANR-19-P3IA-0001 (PRAIRIE 3IA Institute).

References

Alafate Abulimiti, Chloé Clavel, and Justine Cassell. 2023. How about kind of generating hedges using end-to-end neural models? In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics*, Toronto, Canada. Association for Computational Linguistics.

Nalini Ambady and Robert Rosenthal. 1993. Half a minute: Predicting teacher evaluations from thin slices of nonverbal behavior and physical attractiveness. *Journal of personality and social psychology*, 64(3):431.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*.

Jere Brophy. 1981. Teacher praise: A functional analysis. *Review of educational research*, 51(1):5–32.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Valerian J Derlega, Sandra Metts, Sandra Petronio, and Stephen T Margulis. 1993. *Self-disclosure.* Sage Publications, Inc.

Bruce Fraser. 2010. Pragmatic competence: The case of hedging. new approaches to hedging.

Lucie Galland, Catherine Pelachaud, and Florian Pecune. 2022. Adapting conversational strategies to co-optimize agent's task performance and user's engagement. In *Proceedings of the 22nd ACM International Conference on Intelligent Virtual Agents*, pages 1–3.

Pranav Goel, Yoichi Matsuyama, Michael Madaio, and Justine Cassell. 2019. i think it might help if we multiply, and not add. In *Detecting indirectness in conversation. In 9th International Workshop on Spoken Dialogue System Technology*, page 27–40. Springer.

Hector Gómez-Gauchía, Belén Díaz-Agudo, and Pedro A González-Calero. 2006. Conversational strategies in cobber: an affective ccbr framework. *Journal of Experimental & Theoretical Artificial Intelligence*, 18(4):449–469.

Sergiu Hart. 1989. *Shapley value*. Springer.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Dan Jurafsky. 1997. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *www. dcs. shef. ac. uk/nlp/amities/files/bib/ics-tr-97-02. pdf*.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

George Lakoff. 1975. Hedges: A study in meaning criteria and the logic of fuzzy concepts. In *Contemporary research in philosophical logic and linguistic semantics*, pages 221–271. Springer.

Wenqiang Lei, Xiangnan He, Yisong Miao, Qingyun Wu, Richang Hong, Min-Yen Kan, and Tat-Seng Chua. 2020. Estimation-action-reflection: Towards deep interaction between conversational and recommender systems. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 304–312.

Qintong Li, Hongshen Chen, Zhaochun Ren, Pengjie Ren, Zhaopeng Tu, and Zhumin Chen. 2020. Empdg: Multi-resolution interactive empathetic dialogue generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4454–4466.

Zhaojiang Lin, Andrea Madotto, Jamin Shin, Peng Xu, and Pascale Fung. 2019. Moel: Mixture of empathetic listeners. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 121–132.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Michael Madaio, Justine Cassell, and Amy Ogan. 2017a. The impact of peer tutors' use of indirect feedback and instructions. Philadelphia, PA: International Society of the Learning Sciences.

Michael Madaio, Justine Cassell, and Amy Ogan. 2017b. "i think you just got mixed up": confident peer tutors hedge to support partners' face needs. *International Journal of Computer-Supported Collaborative Learning*, 12(4):401–421.

Michael Madaio, Rae Lasko, Amy Ogan, and Justine Cassell. 2017c. Using temporal association rule mining to predict dyadic rapport in peer tutoring. *International Educational Data Mining Society*.

Michael Madaio, Kun Peng, Amy Ogan, and Justine Cassell. 2018. A climate of support: a process-oriented analysis of the impact of rapport on peer tutoring. International Society of the Learning Sciences, Inc.[ISLS].

Michael A Madaio, Amy Ogan, and Justine Cassell. 2016. The effect of friendship and tutoring roles on reciprocal peer tutoring strategies. In *Intelligent Tutoring Systems: 13th International Conference, ITS 2016, Zagreb, Croatia, June 7-10, 2016. Proceedings 13*, pages 423–429. Springer.

Navonil Majumder, Pengfei Hong, Shanshan Peng, Jiankun Lu, Deepanway Ghosal, Alexander Gelbukh, Rada Mihalcea, and Soujanya Poria. 2020. Mime: Mimicking emotions for empathetic response generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8968–8979.

Utku Norman, Tanvi Dinkar, Barbara Bruno, and Chloé Clavel. 2022. Studying alignment in a collaborative learning activity via automatic methods: The link between what we say and do. *Dialogue & Discourse*, 13(2):1–48.

OpenAI. 2022. Chatgpt: Optimizing language models for dialogue.

OpenAI. 2023. Gpt-4.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Natalie Parde and Rodney Nielsen. 2017. Finding patterns in noisy crowds: Regression-based annotation aggregation for crowdsourced data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1907–1912.

Vipul Raheja and Joel Tetreault. 2019. Dialogue act classification with context-aware self-attention. 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 3727–3733.

Yann Raphalen, Chloé Clavel, and Justine Cassell. 2022. ”You might think about slightly revising the title”: Identifying hedges in peer-tutoring interactions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2174, Dublin, Ireland. Association for Computational Linguistics.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Oscar J Romero, Ran Zhao, and Justine Cassell. 2017. Cognitive-inspired conversational-strategy reasoner for socially-aware agents. In *IJCAI*, pages 3807–3813. Melbourne, VIC.

John R Searle. 1965. What is a speech act. *Perspectives in the philosophy of language: a concise anthology*, 2000:253–268.

Jamin Shin, Peng Xu, Andrea Madotto, and Pascale Fung. 2020. Generating empathetic responses by looking ahead the user's sentiment. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7989–7993. IEEE.

Helen Spencer-Oatey. 2005. (im)politeness, face and perceptions of rapport: Unpackaging their bases and interrelationships. 1(1):95–119.

Mukund Sundararajan and Amir Najmi. 2020. The many shapley values for model explanation. In *International conference on machine learning*, pages 9269–9278. PMLR.

Linda Tickle-Degnen and Robert Rosenthal. 1990. The nature of rapport and its nonverbal correlates. *Psychological inquiry*, 1(4):285–293.

Veronika Vincze. 2014. Uncertainty detection in natural language texts. *PhD, University of Szeged*, 141.

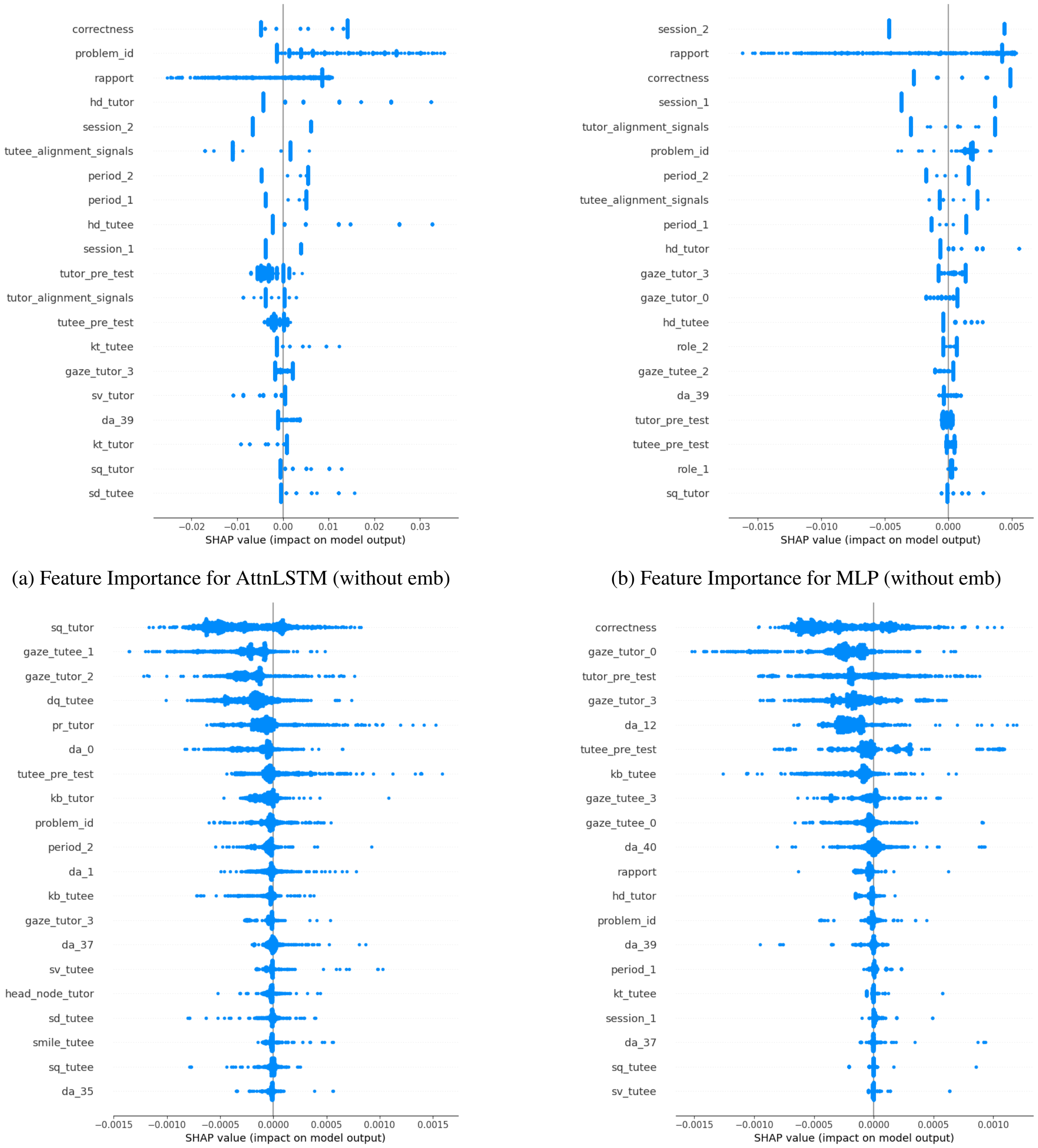
Timothy Williamson. 2002. *Vagueness*. Routledge.

Zhou Yu, Ziyu Xu, Alan W Black, and Alexander Rudnicky. 2016. Strategy and policy learning for non-task-oriented conversational systems. In *Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue*, pages 404–412.

Ran Zhao, Alexandros Papangelis, and Justine Cassell. 2014. Towards a dyadic computational model of rapport management for human-virtual agent interaction. In *International conference on intelligent virtual agents*, pages 514–527. Springer.

Appendix: SHAP Value Graphs

The vertical axis indicates the mean contribution of the feature over the model decision. The horizontal axis indicates how the distribution of features influences the model decision.



(a) Feature Importance for AttnLSTM (without emb)

(b) Feature Importance for MLP (without emb)

(c) Feature Importance for XGBoost

(d) Feature Importance for LightGBM

Figure 2: Feature Importance for Different Models