Edu-ConvoKit
An Open-Source Library for Education Conversation Data

Rose E. Wang
Stanford University
rewang@cs.stanford.edu

Dorottya Demszky
Stanford University
ddemszky@stanford.edu

Abstract

We introduce `Edu-ConvoKit`, an open-source library designed to handle pre-processing, annotation and analysis of conversation data in education. Resources for analyzing education conversation data are scarce, making the research challenging to perform and therefore hard to access. We address these challenges with `Edu-ConvoKit`. `Edu-ConvoKit` is open-source[1], pip-installable[2], with comprehensive documentation[3]. Our demo video is available at: https://youtu.be/zdcI839vAko?si=h9qlnl76ucSuXb8-. We include additional resources, such as Colab applications of `Edu-ConvoKit` to three diverse education datasets[4] and a repository of `Edu-ConvoKit`-related papers[5].

1 Introduction

Language is central to educational interactions, ranging from classroom instruction to tutoring sessions to peer discussions. It offers rich insights into the teaching and learning process that go beyond the current, oversimplified view of relying on standardized test outcomes (Wentzel, 1997; Pianta et al., 2003; Robinson, 2022; Wentzel, 2022). The landscape of natural language processing (NLP) and education is rapidly evolving, with an increase of open-sourced education conversation datasets (e.g., from Caines et al. (2020); Stasaski et al. (2020); Suresh et al. (2021a); Demszky and Hill (2023); Wang et al. (2023a,c); Holt (2023)), heightened interest manifesting in academic venues (e.g., NeurIPS GAIED (2023), Building Educational Applications at *ACL Conferences BEA (2023), and education conferences hosting NLP tracks[6]), alongside courses dedicated to this field (e.g., Stanford's NLP and Education course CS293[7]).

Challenges and consequences. While the interest in this interdisciplinary field is growing, our conversations with education data science and NLP researchers both in academia and industry have surfaced several challenges that hinder research progress. First, there is **no centralized tool or resource** that assists in analyzing education data, or helps researchers understand different tradeoffs in methods. For example, researchers expressed uncertainty about pre-processing the data, such as "the best way to anonymize the data to protect the privacy of students and teachers". They also wanted an "easily accessible collection of language tools and models that can detect insightful things." The lack of these tools and resources makes the research harder to conduct. Second, there is a **high learning curve for performing computational analyses**. For example, many education researchers are trained in qualitative research; even though they want to use computational tools for quantitative analyses at scale, they often do not know how to start or have the readily available compute to try out the tools.

Our system. Our work introduces `Edu-ConvoKit` to address these challenges. `Edu-ConvoKit` is designed to facilitate and democratize the study of education conversation data. It is a modular, end-to-end pipeline for **A.** pre-processing, **B.** annotating, and **C.** analyzing education conversation data, illustrated in Figure 1. Specifically, `Edu-ConvoKit`

- **Supports pre-processing** for education con-

[1] https://github.com/stanfordnlp/edu-convokit
[2] https://pypi.org/project/edu-convokit/
[3] https://edu-convokit.readthedocs.io/en/latest/
[4] https://github.com/stanfordnlp/edu-convokit?tab=readme-ov-file#datasets-with-edu-convokit
[5] https://github.com/stanfordnlp/edu-convokit/blob/main/papers.md
[6] The International Conference on Learning Analytics and Knowledge (LAK), Education Data Mining (EDM), and Artificial Intelligence in Education (AIED).
[7] https://web.stanford.edu/class/cs293/

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: System Demonstrations), pages 61–69
June 16-21, 2024 ©2024 Association for Computational Linguistics

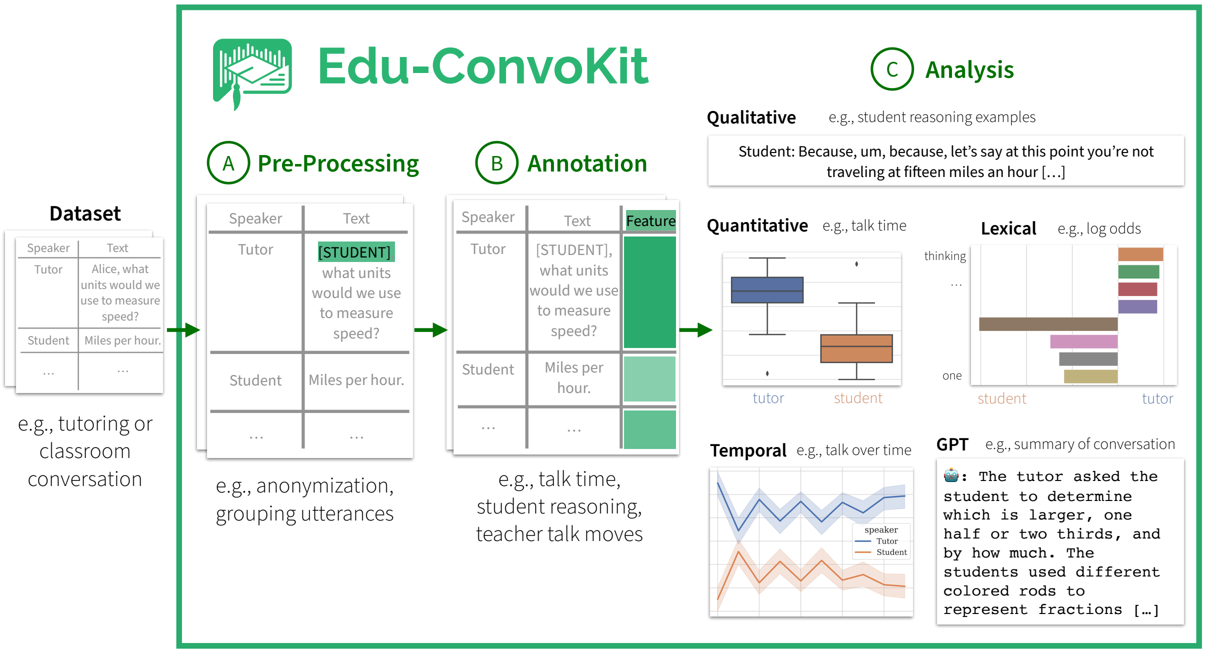

Edu-ConvoKit
Dataset
Speaker
Text
Tutor
Alice, what units would we use to measure speed?
Student
Miles per hour.
...
e.g., tutoring or classroom conversation
A Pre-Processing
Speaker
Text
Tutor
[STUDENT] what units would we use to measure speed?
Student
Miles per hour.
...
e.g., anonymization, grouping utterances
B Annotation
Speaker
Text
Feature
Tutor
[STUDENT], what units would we use to measure speed?
Student
Miles per hour.
...
e.g., talk time, student reasoning, teacher talk moves
C Analysis
Qualitative e.g., student reasoning examples
Student: Because, um, because, let's say at this point you're not traveling at fifteen miles an hour [...]
Quantitative e.g., talk time
tutor
student
Lexical e.g., log odds
thinking
...
one
student
tutor
Temporal e.g., talk over time
speaker
Tutor
Student
GPT e.g., summary of conversation
: The tutor asked the student to determine which is larger, one half or two thirds, and by how much. The students used different colored rods to represent fractions [...]

Figure 1: **Overview of `Edu-ConvoKit`.** `Edu-ConvoKit` is designed to facilitate the study of conversation data in education. It is a modular, end-to-end pipeline for **A.** pre-processing, **B.** annotating, and **C.** analyzing education conversation data. As additional resources, the toolkit includes Colab notebooks applying `Edu-ConvoKit` to three existing, large education datasets and a centralized database of `Edu-ConvoKit` papers. This toolkit aims to enhance the accessibility and reproducibility of NLP and education research.

versation datasets, such as automatically de-identifying conversations;

- **Hosts a collection of language tools and models for annotation**, ranging from traditional (e.g., talk time) to neural measures (e.g., classifying student reasoning); and

- **Automates several analyses** used in NLP and education research, ranging from qualitative analyses, temporal analyses and GPT-powered analyses (e.g., on summarizing transcripts).

To demonstrate its flexible design and ensure its accessibility regardless of compute infrastructure, we created **Colab notebooks of `Edu-ConvoKit` applied to three diverse education conversation datasets** in mathematics (Demszky and Hill, 2023; Suresh et al., 2021b; Holt, 2023). We additionally created **a centralized database of research projects** that have either used `Edu-ConvoKit` or have features integrated in the toolkit. We invite the community to contribute to the toolkit and collectively push the boundaries of education conversation research!

2 Related Works

2.1 Advancing NLP through Toolkits

The NLP community has benefited greatly from the public availability of general toolkits, which standardize the way data is transformed, annotated and analyzed. Examples include NLTK (Bird, 2006), StanfordNLP (Qi et al., 2019), spaCy (Honnibal et al., 2020), or scikit-learn (Pedregosa et al., 2011). They improve the accessibility to the research and allow researchers to focus on developing new methods, rather than on re-implementing existing ones. `Edu-ConvoKit` shares these goals. ConvoKit (Chang et al., 2020) is a NLP package for conversational analysis and bears the most similarity to our work. A key difference between our library and ConvoKit is the data structure: `Edu-ConvoKit` uses a table-based dataframe structure whereas ConvoKit uses an object-based data structure akin to a dictionary. Our data structure makes manipulating data easier, e.g., performing utterance-level annotations. Additionally, our tool caters to education language research and therefore supports an array of common analyses such as qualitative analysis (Erickson et al., 1985; Corbin and Strauss, 1990; Wang et al., 2023b), quantitative evaluations (Bienkowski et al., 2012; Kim and Piech, 2023; Demszky et al., 2023), or lexical comparisons (Praharaj et al., 2021; Handa et al., 2023).

2.2 Supporting the Multifaceted Nature of Education Interaction Research

Edu-ConvoKit sits at the intersection of many disciplines that use different annotation and analysis tools for understanding language use in education interactions. For example, qualitative education research uses *qualitative analysis* to manually analyze the discourse, such as how students collaborate with each other (Mercer, 1996; Jackson et al., 2013; Langer-Osuna et al., 2020; Chen, 2020; Hunkins et al., 2022). Learning analytics uses *quantitative and temporal analysis* to summarize statistics in aggregate or over time (Bienkowski et al., 2012; Kim and Piech, 2023; Demszky et al., 2023, 2024). Other areas perform *lexical analyses and neural measures for annotating* education discourse features (Reilly and Schneider, 2019; Praharaj et al., 2021; Rahimi et al., 2017; Alic et al., 2022; Hunkins et al., 2022; Demszky and Hill, 2023; Reitman et al., 2023; Suresh et al., 2021a; Himmelsbach et al., 2023; Wang and Demszky, 2023). Recently, newer analysis tools powered by GPT models analyze complete conversations such as summarizing or pulling good examples of teacher instruction from the classroom transcripts (Wang and Demszky, 2023). Edu-ConvoKit is designed to support these forms of annotation and analysis, and unify the currently fragmented software ecosystem of this interdisciplinary research area.

3 Design Principles

Edu-ConvoKit follows these principles:

I. Minimalistic Data Structure. The system transforms all data inputs (e.g., csv and json files) into a dataframe. Edu-ConvoKit only needs the speaker and text columns to be uniquely identifiable, which is the case in the datasets we surveyed and applied Edu-ConvoKit to.

II. Efficient Execution. The system should be able to run on a CPU and support large-scale pre-processing, annotation and analysis.

III. Modularity. Each component of Edu-ConvoKit functions as an independent module. Running one module (e.g., pre-processing) should not be required for the user to run another module (e.g., annotation).

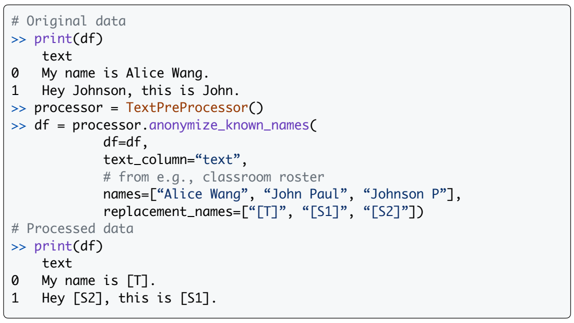
These principles enable Edu-ConvoKit to comprehensively incorporate different methods for preprocessing, annotation and analysis. They ensure that Edu-ConvoKit is effective and adaptable to various research needs.

4 Edu-ConvoKit

Edu-ConvoKit is organized around three entities: PreProcessor, Annotator, and Analyzer (see Figure 1). The following sections enumerate each entity's functionality. Please refer to the short demo video to preview Edu-ConvoKit in action: https://youtu.be/zdcI839vAko?si=h9qlnl76ucSuXb8-.

4.1 PreProcessor

The PreProcessor module in Edu-ConvoKit processes the raw data and includes several techniques standard to education and NLP research practices, such as replacing speaker names with unique identifiers, merging consecutive utterances by the same speaker, and formatting text to be human-readable. Figure 3 illustrates a simple example of text de-identification with PreProcessor, assuming that the researcher has access to a list of names (e.g. classroom roster) to be replaced. PreProcessor accounts for multiple names per individual, and users can define how each name should be replaced. This feature ensures that the context of each interaction is preserved while maintaining confidentiality of the participants.

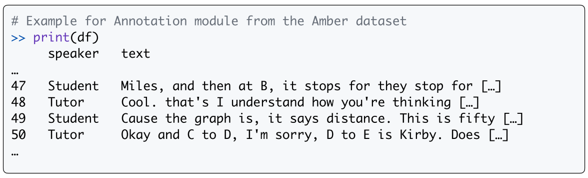


```
# Original data
>> print(df)
    text
0   My name is Alice Wang.
1   Hey Johnson, this is John.
>> processor = TextPreProcessor()
>> df = processor.anonymize_known_names(
            df=df,
            text_column=“text”,
            # from e.g., classroom roster
            names=[“Alice Wang”, “John Paul”, “Johnson P”],
            replacement_names=[“[T]”, “[S1]”, “[S2]”])
# Processed data
>> print(df)
    text
0   My name is [T].
1   Hey [S2], this is [S1].
```

Figure 2: **Example for text de-identification.** PreProcessor accounts for multiple names (e.g., “John Paul” matches to “John”), and handles word boundaries (e.g., “John” does not match to “Johnson”).

4.2 Annotator

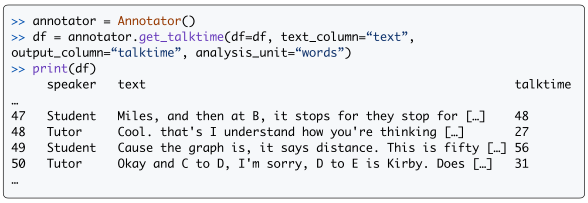
Annotator annotates features at an *utterance-level*. It currently supports 7 types of features, ranging from traditional to neural measures of educational discourse. The features follow the original implementations of cited works and the neural measures are models hosted on HuggingFace hub. Notably, `Annotator` performs annotation with a single function call. The following sections describe these features, using Figure 3 as the running example.



```
# Example for Annotation module from the Amber dataset
>> print(df)
     speaker   text
…
47   Student   Miles, and then at B, it stops for they stop for […]
48   Tutor     Cool. that's I understand how you're thinking […]
49   Student   Cause the graph is, it says distance. This is fifty […]
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]
…
```

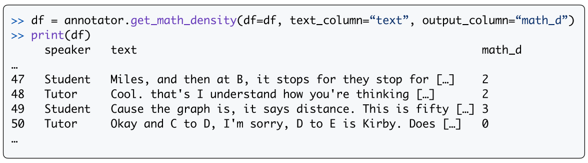
Figure 3: **Example for `Annotator`**.

Talk Time. Talk time measures the amount of speaker talk by word count and timestamps (if provided in the dataset). This feature quantifies the participation of both teachers/tutors and students, offering insights into classroom dynamics (TeachFX; Jensen et al., 2020; Demszky et al., 2024).



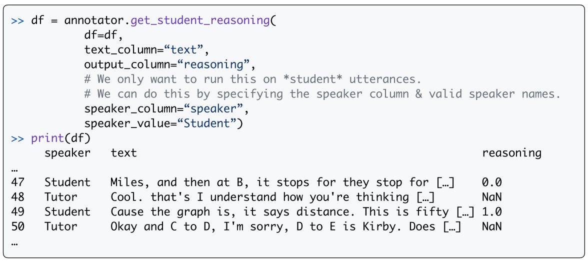
```
>> annotator = Annotator()
>> df = annotator.get_talktime(df=df, text_column="text",
output_column="talktime", analysis_unit="words")
>> print(df)
     speaker   text                                                     talktime
…
47   Student   Miles, and then at B, it stops for they stop for […]    48
48   Tutor     Cool. that's I understand how you're thinking […]       27
49   Student   Cause the graph is, it says distance. This is fifty […] 56
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   31
…
```

Math Density. Math density measures the number of math terms used in an utterance, where the dictionary of math terms was collected in prior work by mathematics education researchers (Himmelsbach et al., 2023). This feature provides a quantitative measure of mathematical content in the dialogue.



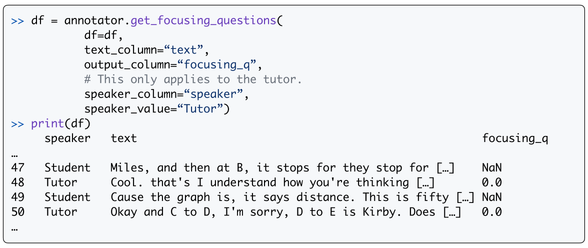
```
>> df = annotator.get_math_density(df=df, text_column="text", output_column="math_d")
>> print(df)
     speaker   text                                                     math_d
…
47   Student   Miles, and then at B, it stops for they stop for […]    2
48   Tutor     Cool. that's I understand how you're thinking […]       2
49   Student   Cause the graph is, it says distance. This is fifty […] 3
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   0
…
```

Student Reasoning. The student reasoning annotation measures whether a given student utterance provides a mathematical explanation for an idea, procedure or solution (Demszky and Hill, 2023; Hill et al., 2008). The model is a finetuned RoBERTa classifier (Liu et al., 2019) on instances of student reasoning from elementary math classroom transcripts. `Edu-ConvoKit` follows the original implementation from Demszky and Hill (2023), ensuring fidelity to prior research: `Annotator` only label utterances that are at least 8 words long based on word boundaries; all other utterances are annotated as NaN. Furthermore, users can also easily specify which speakers to annotate for, such as to only annotate the student speakers as shown in the example below.



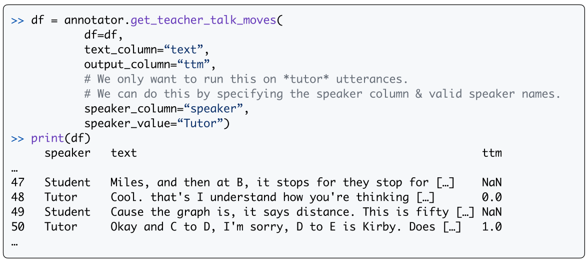
```
>> df = annotator.get_student_reasoning(
          df=df,
          text_column="text",
          output_column="reasoning",
          # We only want to run this on *student* utterances.
          # We can do this by specifying the speaker column & valid speaker names.
          speaker_column="speaker",
          speaker_value="Student")
>> print(df)
     speaker   text                                                     reasoning
…
47   Student   Miles, and then at B, it stops for they stop for […]    0.0
48   Tutor     Cool. that's I understand how you're thinking […]       NaN
49   Student   Cause the graph is, it says distance. This is fifty […] 1.0
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   NaN
…
```

Focusing Questions. The focusing question annotation capture questions that attend to what the student is thinking and presses them to communicate their thoughts clearly (Leinwand et al., 2014; Alic et al., 2022). The model is a finetuned RoBERTa classifier (Liu et al., 2019) on instances of teacher focusing questions from elementary math classroom transcripts:



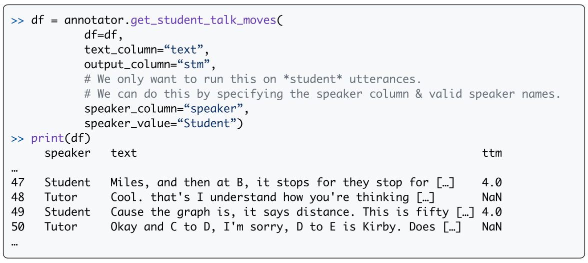
```
>> df = annotator.get_focusing_questions(
          df=df,
          text_column="text",
          output_column="focusing_q",
          # This only applies to the tutor.
          speaker_column="speaker",
          speaker_value="Tutor")
>> print(df)
     speaker   text                                                     focusing_q
…
47   Student   Miles, and then at B, it stops for they stop for […]    NaN
48   Tutor     Cool. that's I understand how you're thinking […]       0.0
49   Student   Cause the graph is, it says distance. This is fifty […] NaN
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   0.0
…
```

Teacher Accountable Talk Moves. Teacher accountable talk moves capture the teacher's strategies to promote equitable participation in classrooms (Suresh et al., 2021b; Jacobs et al., 2022), based on the Accountable Talk framework (O'Connor et al., 2015). It is a finetuned ELECTRA 7-way classifier (Clark et al., 2020) where: 0: No Talk Move Detected, 1: Keeping Everyone Together, 2: Getting Students to Related to Another Student's Idea, 3: Restating, 4: Revoicing, 5: Pressing for Accuracy, 6: Pressing for Reasoning.



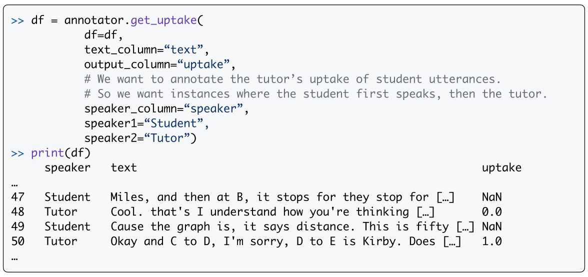
```
>> df = annotator.get_teacher_talk_moves(
          df=df,
          text_column="text",
          output_column="ttm",
          # We only want to run this on *tutor* utterances.
          # We can do this by specifying the speaker column & valid speaker names.
          speaker_column="speaker",
          speaker_value="Tutor")
>> print(df)
     speaker   text                                                     ttm
…
47   Student   Miles, and then at B, it stops for they stop for […]    NaN
48   Tutor     Cool. that's I understand how you're thinking […]       0.0
49   Student   Cause the graph is, it says distance. This is fifty […] NaN
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   1.0
…
```

Student Accountable Talk Moves. Analogous to the teacher talk moves, the student accountable talk moves are student discussion strategies to promote equitable participation in a rigorous classroom learning environment (Suresh et al., 2021b; Jacobs et al., 2022). It is also a finetuned ELECTRA classifier for 5 classes: 0: No Talk Move Detected, 1: Relating to Another Student, 2: Asking for More Information, 3: Making a Claim, 4: Providing Evidence or Reasoning.



```
>> df = annotator.get_student_talk_moves(
            df=df,
            text_column="text",
            output_column="stm",
            # We only want to run this on *student* utterances.
            # We can do this by specifying the speaker column & valid speaker names.
            speaker_column="speaker",
            speaker_value="Student")
>> print(df)
     speaker   text                                                ttm
…
47   Student   Miles, and then at B, it stops for they stop for […]    4.0
48   Tutor     Cool. that's I understand how you're thinking […]       NaN
49   Student   Cause the graph is, it says distance. This is fifty […] 4.0
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   NaN
…
```

Conversational Uptake. Conversational uptake measures how teachers build on the contributions of students (Demszky et al., 2021). It is a BERT model fine-tuned with a self-supervised training objective (next utterance prediction), on an elementary math classroom dataset (Demszky and Hill, 2023), Switchboard (Godfrey and Holliman, 1997) and a tutoring dataset. `Annotator` annotates utterances according to the original implementation: It can label teacher utterances following substantive student utterances that are at least 5 words long, such as in the example below.

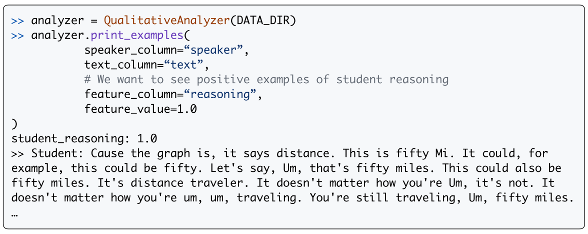


```
>> df = annotator.get_uptake(
            df=df,
            text_column="text",
            output_column="uptake",
            # We want to annotate the tutor’s uptake of student utterances.
            # So we want instances where the student first speaks, then the tutor.
            speaker_column="speaker",
            speaker1="Student",
            speaker2="Tutor")
>> print(df)
     speaker   text                                                uptake
…
47   Student   Miles, and then at B, it stops for they stop for […]    NaN
48   Tutor     Cool. that's I understand how you're thinking […]       0.0
49   Student   Cause the graph is, it says distance. This is fifty […] NaN
50   Tutor     Okay and C to D, I'm sorry, D to E is Kirby. Does […]   1.0
…
```

4.3 `Analyzer`

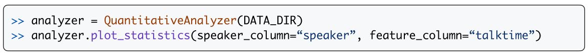
`Edu-ConvoKit` supports several modules that cover common analyses in education conversation research. In general, each module is exposed by three methods: `plot` for plotting, `print` for displaying results in the terminal, and `report` for outputting results as text. There are multiple data entry points for the `Analyzer` such as a single or multiple transcripts, or a data directory. The following sections describe these modules, assuming that the variable `DATA_DIR` is a directory of annotated transcripts.

`QualitativeAnalyzer`. This module enables researchers to view annotation examples. For example, we can easily view positive examples of student reasoning below. This module has other features, such as additionally showing the previous and subsequent lines around the examples; please refer to our documentation for all features.

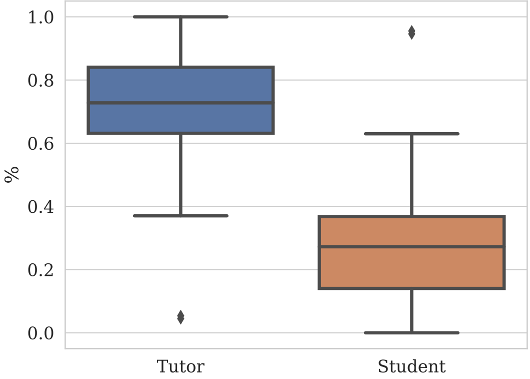


```
>> analyzer = QualitativeAnalyzer(DATA_DIR)
>> analyzer.print_examples(
            speaker_column="speaker",
            text_column="text",
            # We want to see positive examples of student reasoning
            feature_column="reasoning",
            feature_value=1.0
)
student_reasoning: 1.0
>> Student: Cause the graph is, it says distance. This is fifty Mi. It could, for
example, this could be fifty. Let's say, Um, that's fifty miles. This could also be
fifty miles. It's distance traveler. It doesn't matter how you're Um, it's not. It
doesn't matter how you're um, um, traveling. You're still traveling, Um, fifty miles.
…
```

`QuantitativeAnalyzer`. This module reports the quantitative summaries of the annotation results. Users can also flexibly group and use different representations, such as grouping by speaker or displaying the values as percentages as shown below.

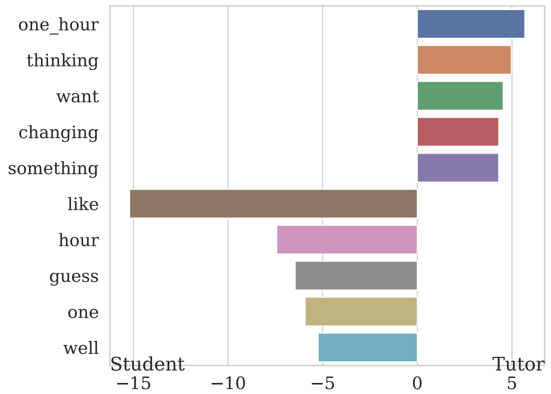


```
>> analyzer = QuantitativeAnalyzer(DATA_DIR)
>> analyzer.plot_statistics(speaker_column="speaker", feature_column="talktime")
```



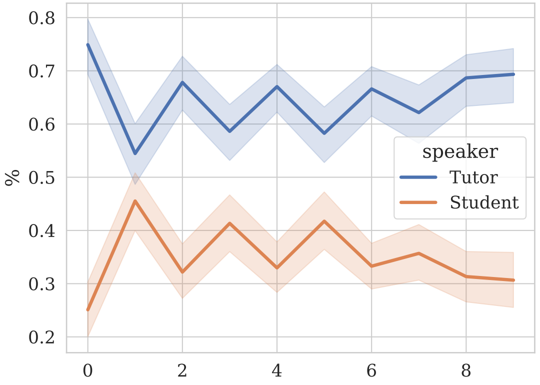
`LexicalAnalyzer`. This module reports language patterns on the word-level. It can report n-gram frequency and weighted log-odds analysis from Section 3.4 of Monroe et al. (2008), which reports which n-grams are more likely to be uttered by one group over the other given a prior distribution of words; currently, the priors are defined based on the provided dataset, however we hope to flexibly handle any user-provided priors in the future. Below is an example of the log-odds analysis that shows the top 5 n-grams in the student's utterances over the tutor's.

```
>> analyzer = LexicalAnalyzer()
>> df1 = df[df[“speaker”] == “Student”]
>> df2 = df[df[“speaker”] == “Tutor”]
>> analyzer.plot_logodds(df1=df1, df2=df2, text_column1=“text”, text_column2=“text”,
topk=5)
```



TemporalAnalyzer. This module provides a time analysis of the annotations over the course of the conversation(s). Similar to `QuantitativeAnalyzer`, it can group and report the data in different ways. An important variable to this module is `num_bins`, which indicates how many time bins the transcript should be split into; currently, the split is based on transcript lines, however we hope to support other split criteria in the future such as by word count. Below is an example with speaker talk time.

```
>> analyzer = TemporalAnalyzer(DATA_DIR)
>> analyzer.plot_statistics(speaker_column=“speaker”, feature_column=“talktime”,
  num_bins=10)
```



GPTConversationAnalyzer. This module uses GPT models accessible through the OpenAI API to analyze on the conversation-level with natural language. Some prompts include summarizing the conversation (below example) or generating suggestions to the teacher/tutor on eliciting more student reasoning from Wang and Demszky (2023). The module has additional features (not shown) such as automatically truncating the transcript if it surpasses the model's context length, adding line numbers to the conversation or altering how the lines should be formatted.

```
>> analyzer = GPTConversationAnalyzer()
>> prompt, response = analyzer.run_prompt(df=df, prompt_name=“summarize”,
speaker_column=“speaker”, text_column=“text”, model=“gpt-4”)
>> print(response)
In this conversation, a tutor guides a student through a virtual learning platform,
explaining various tools and how to use them, such as the drawing tool, text box, and
erasure function. They then apply these tools to a math problem about measuring the
distance and time taken from Providence to Newark. The student engages in problem-
solving, plotting points on a map, and interpreting a graph, while the tutor
encourages thinking aloud and self-questioning. The tutor emphasizes understanding
the relationship between variables like distance, time, and speed. […]
```

5 Additional Resources: Basic Tutorials, Case Studies, and Paper Repository

We create a suite of introductory tutorials and case studies of `Edu-ConvoKit` as Colab notebooks (link). To demonstrate its wide applicability and generalizable design structure, we apply `Edu-ConvoKit` to three different education transcript datasets developed by different authors: NCTE, an elementary school classroom dataset (Demszky and Hill, 2023); TalkMoves, a K-12 classroom dataset (Suresh et al., 2021b); and Amber, a one-on-one 8th-9th grade tutoring dataset (Holt, 2023). For space reasons, we omit the findings of the case studies in this paper, but they can be found in our GitHub repository. To centralize research efforts, we additionally contribute a paper repository that include papers that have used `Edu-ConvoKit` or have features incorporated into `Edu-ConvoKit` (link).

6 Conclusion

We introduce `Edu-ConvoKit`, an open-source library designed to democratize and enhance the study of education conversation data. Implemented in Python and easily accessible via GitHub and pip installation, it offers a user-friendly interface complete with extensive documentation, tutorials, applications to three diverse education datasets, and paper repository resource. Based on extensive research experience, it incorporates best practices for pre-processing data and a series of different annotation measures grounded in prior literature, such as measuring student reasoning and talk time. It additionally supports several analysis modules, such as temporal analyses (e.g., talk time ratios), lexical analyses (e.g., word usage) and GPT-powered analyses (e.g., summarization). Fostering a collaborative environment through its open-source nature, `Edu-ConvoKit` and its resources unify research efforts in this exciting interdisciplinary field to improve teaching and learning.

7 Limitations

There are limitations to Edu-ConvoKit which we intend on addressing in future versions of the library. Some of the current limitations include: Edu-ConvoKit does not support transcription; it does not support connecting the language analyses to metadata, such as demographic data or learning outcomes, such as in Demszky and Hill (2023); it only supports English-focused annotation methods; many of its annotation models were trained on elementary and middle school mathematics, so they may not generalize to other domains; and Edu-ConvoKit's de-identification method assumes the speakers are known. There are other existing de-identification methods that do not assume knowledge of the speaker names (one of which is also implemented in Edu-ConvoKit) however these methods are known to have high false-negative and false-positive rates.

8 Ethics Statement

The intended use case for this toolkit is to further education research and improve teaching and learning outcomes through the use of NLP techniques. Edu-ConvoKit is intended for research purposes only. Edu-ConvoKit uses data from existing public datasets that acquired consent from parents and teachers when applicable; for example, the NCTE dataset from Demszky and Hill (2023) acquired consent from parents and teachers for their study (Harvard's IRB #17768), and for the de-identified data to be publicly shared. As stewards of this library which builds on these datasets, we are committed to protecting the confidentiality of the individuals and ask users of our library to do the same. It is important to note that inferences drawn using Edu-ConvoKit may not necessarily reflect generalizable observations (e.g., the student reasoning model was trained on elementary school math, and may not yield correct insights when applied to high school math). Therefore, the analysis results should be interpreted with caution. Unacceptable use cases include any attempts to identify users or use the data for commercial gain. We additionally recommend that researchers who do use our toolkit take steps to mitigate any risks or harms to individuals that may arise.

Acknowledgements

We are thankful to Chris Manning, Omar Khattab, Ali Malik, Jim Malamut, Lucy Li for their feedback on the work. Additionally, we are thankful to Yann Hicke for their contributions to Edu-ConvoKit.

References

Sterling Alic, Dorottya Demszky, Zid Mancenido, Jing Liu, Heather Hill, and Dan Jurafsky. 2022. Computationally identifying funneling and focusing questions in classroom discourse. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 224–233, Seattle, Washington. Association for Computational Linguistics.

BEA. 2023. Workshop on Innovative Use of NLP for Building Educational Applications . https://sig-edu.org/bea/2024. [Online; accessed 20-Dec-2023].

Marie Bienkowski, Mingyu Feng, and Barbara Means. 2012. Enhancing teaching and learning through educational data mining and learning analytics: An issue brief. *Office of Educational Technology, US Department of Education*.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Andrew Caines, Helen Yannakoudakis, Helena Edmondson, Helen Allen, Pascual Pérez-Paredes, Bill Byrne, and Paula Buttery. 2020. The teacher-student chatroom corpus. In *Proceedings of the 9th Workshop on NLP for Computer Assisted Language Learning*, pages 10–20.

Jonathan P Chang, Caleb Chiam, Liye Fu, Andrew Z Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. Convokit: A toolkit for the analysis of conversations. *arXiv preprint arXiv:2005.04246*.

Gaowei Chen. 2020. A visual learning analytics (vla) approach to video-based teacher professional development: Impact on teachers' beliefs, self-efficacy, and classroom talk practice. *Computers & Education*, 144:103670.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Juliet M Corbin and Anselm Strauss. 1990. Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative sociology*, 13(1):3–21.

Dorottya Demszky and Heather Hill. 2023. The NCTE transcripts: A dataset of elementary math classroom transcripts. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 528–538, Toronto, Canada. Association for Computational Linguistics.

Dorottya Demszky, Jing Liu, Heather C Hill, Shyamoli Sanghi, and Ariel Chung. 2023. Improving teachers' questioning quality through automated feedback: A mixed-methods randomized controlled trial in brick-and-mortar classrooms.

Dorottya Demszky, Jing Liu, Zid Mancenido, Julie Cohen, Heather Hill, Dan Jurafsky, and Tatsunori Hashimoto. 2021. Measuring conversational uptake: A case study on student-teacher interactions. 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 1638–1653, Online. Association for Computational Linguistics.

Dorottya Demszky, Rose Wang, Sean Geraghty, and Carol Yu. 2024. Does feedback on talk time increase student engagement? evidence from a randomized controlled trial on a math tutoring platform.

Frederick Erickson et al. 1985. *Qualitative methods in research on teaching*. Institute for Research on Teaching.

GAIED. 2023. NeurIPS'23 Workshop: Generative AI for Education (GAIED). `https://gaied.org/neurips2023/`. [Online; accessed 20-Dec-2023].

John J Godfrey and Edward Holliman. 1997. Switchboard-1 release 2. *Linguistic Data Consortium, Philadelphia*, 926:927.

Kunal Handa, Margarett Clapper, Jessica Boyle, Rose Wang, Diyi Yang, David Yeager, and Dorottya Demszky. 2023. "mistakes help us grow": Facilitating and evaluating growth mindset supportive language in classrooms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8877–8897.

Heather C Hill, Merrie L Blunk, Charalambos Y Charalambous, Jennifer M Lewis, Geoffrey C Phelps, Laurie Sleep, and Deborah Loewenberg Ball. 2008. Mathematical knowledge for teaching and the mathematical quality of instruction: An exploratory study. *Cognition and instruction*, 26(4):430–511.

Zachary Himmelsbach, Heather C. Hill, Jing Liu, and Dorottya Demszky. 2023. A quantitative study of mathematical language in classrooms. *EdWorkingPapers*.

Laurence Holt. 2023. xq-data.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, Adriane Boyd, et al. 2020. spacy: Industrial-strength natural language processing in python.

Nicholas Hunkins, Sean Kelly, and Sidney D'Mello. 2022. "beautiful work, you're rock stars!": Teacher analytics to uncover discourse that supports or undermines student motivation, identity, and belonging in classrooms. In *LAK22: 12th International Learning Analytics and Knowledge Conference*, pages 230–238.

Kara Jackson, Anne Garrison, Jonee Wilson, Lynsey Gibbons, and Emily Shahan. 2013. Exploring relationships between setting up complex tasks and opportunities to learn in concluding whole-class discussions in middle-grades mathematics instruction. *Journal for Research in Mathematics Education*, 44(4):646–682.

Jennifer Jacobs, Karla Scornavacco, Charis Harty, Abhijit Suresh, Vivian Lai, and Tamara Sumner. 2022. Promoting rich discussions in mathematics classrooms: Using personalized, automated feedback to support reflection and instructional change. *Teaching and Teacher Education*, 112:103631.

Emily Jensen, Meghan Dale, Patrick J Donnelly, Cathlyn Stone, Sean Kelly, Amanda Godley, and Sidney K D'Mello. 2020. Toward automated feedback on teacher discourse to enhance teacher learning. In *Proceedings of the 2020 chi conference on human factors in computing systems*, pages 1–13.

Yunsung Kim and Chris Piech. 2023. High-resolution course feedback: Timely feedback mechanism for instructors. In *Proceedings of the Tenth ACM Conference on Learning@ Scale*, pages 81–91.

Jennifer Langer-Osuna, Jen Munson, Emma Gargroetzi, Immanuel Williams, and Rosa Chavez. 2020. "so what are we working on?": How student authority relations shift during collaborative mathematics activity. *Educational Studies in Mathematics*, 104:333–349.

SE Leinwarnd et al. 2014. National council of teachers of mathematics. *Principles ro actions: Ensuring Mathematical success for all. Reston: VA: Author*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Neil Mercer. 1996. The quality of talk in children's collaborative activity in the classroom. *Learning and instruction*, 6(4):359–377.

Burt L Monroe, Michael P Colaresi, and Kevin M Quinn. 2008. Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403.

Catherine O'Connor, Sarah Michaels, and Suzanne Chapin. 2015. *"Scaling Down" to Explore the Role of Talk in Learning: From District Intervention to Controlled Classroom Study*, pages 111–126.

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.

Robert C Pianta, Bridget Hamre, and Megan Stuhlman. 2003. Relationships between teachers and children.

Sambit Praharaj, Maren Scheffel, Marcel Schmitz, Marcus Specht, and Hendrik Drachsler. 2021. Towards automatic collaboration analytics for group speech data using learning analytics. *Sensors*, 21(9):3156.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D Manning. 2019. Universal dependency parsing from scratch. *arXiv preprint arXiv:1901.10457*.

Zahra Rahimi, Diane Litman, Richard Correnti, Elaine Wang, and Lindsay Clare Matsumura. 2017. Assessing students' use of evidence and organization in response-to-text writing: Using natural language processing for rubric-based automated scoring. *International Journal of Artificial Intelligence in Education*, 27(4):694–728.

Joseph M Reilly and Bertrand Schneider. 2019. Predicting the quality of collaborative problem solving through linguistic analysis of discourse. *International Educational Data Mining Society*.

Jason G Reitman, Charis Clevenger, Quinton Beck-White, Amanda Howard, Sierra Rose, Jacob Elick, Julianna Harris, Peter Foltz, and Sidney K D'Mello. 2023. A multi-theoretic analysis of collaborative discourse: A step towards ai-facilitated student collaborations. In *International Conference on Artificial Intelligence in Education*, pages 577–589. Springer.

Carly D Robinson. 2022. A framework for motivating teacher-student relationships. *Educational Psychology Review*, 34(4):2061–2094.

Katherine Stasaski, Kimberly Kao, and Marti A. Hearst. 2020. CIMA: A large open access dialogue dataset for tutoring. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–64, Seattle, WA, USA â†' Online. Association for Computational Linguistics.

A. Suresh, J. Jacobs, V. Lai, C. Tan, W. Ward, J. H. Martin, and T. Sumner. 2021a. Using transformers to provide teachers with personalized feedback on their classroom discourse: The talkmoves application. arxiv. Preprint.

A Suresh, J Jacobs, V Lai, C Tan, W Ward, JH Martin, and T Sumner. 2021b. Using transformers to provide teachers with personalized feedback on their classroom discourse: The talkmoves application. *Association for the Advancement of Artificial Intelligence*.

TeachFX. Teachfx.

Rose Wang and Dorottya Demszky. 2023. Is ChatGPT a good teacher coach? measuring zero-shot performance for scoring and providing actionable insights on classroom instruction. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 626–667, Toronto, Canada. Association for Computational Linguistics.

Rose Wang, Pawan Wirawarn, Noah Goodman, and Dorottya Demszky. 2023a. Sight: A large annotated dataset on student insights gathered from higher education transcripts. In *Proceedings of Innovative Use of NLP for Building Educational Applications*.

Rose Wang, Pawan Wirawarn, Noah Goodman, and Dorottya Demszky. 2023b. SIGHT: A large annotated dataset on student insights gathered from higher education transcripts. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 315–351, Toronto, Canada. Association for Computational Linguistics.

Rose E. Wang, Qingyang Zhang, Carly Robinson, Susanna Loeb, and Dorottya Demszky. 2023c. Step-by-step remediation of students' mathematical mistakes.

Kathryn R Wentzel. 1997. Student motivation in middle school: The role of perceived pedagogical caring. *Journal of educational psychology*, 89(3):411.

Kathryn R Wentzel. 2022. Does anybody care? conceptualization and measurement within the contexts of teacher-student and peer relationships. *Educational Psychology Review*, pages 1–36.