Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2023"

June 14–16, 2023

Linguistic Annotation Generation with ChatGPT: a Synthetic Dataset of Speech Functions for Discourse Annotation of Casual Conversations

Lidiia Ostyakova[♡,◇] ostyakova.ln@gmail.com
Kseniia Petukhova[◇] petukhova.ka@mipt.ru
Veronika Smilga [◇] smilgaveronika@gmail.com
Dilyara Zharikova[◇] dilyara.rimovna@gmail.com

[♡]HSE University
[◇]Moscow Institute of Physics and Technology

Abstract

This paper is devoted to examining the hierarchical and multilayered taxonomy of Speech Functions, encompassing pragmatics, turn-taking, feedback, and topic switching in open-domain conversations. To evaluate the distinctiveness of closely related pragmatic classes, we conducted comparative analyses involving both expert annotators and crowdsourcing workers. We then carried out classification experiments on a manually annotated dataset and a synthetic dataset generated using ChatGPT. We looked into the viability of using ChatGPT to produce data for such complex topics as discourse. Our findings contribute to the field of prompt engineering techniques for linguistic annotation in large language models, offering valuable insights for the development of more sophisticated dialogue systems.

Keywords: speech functions, ChatGPT, dialogue systems, discourse analysis, open-domain conversations
DOI: 10.28995/2075-7182-2023-22-386-403

Генерация лингвистических данных с помощью ChatGPT: создание синтетического корпуса речевых функций для разметки дискурса в диалогах на повседневные темы

Лидия Остякова[♡,◇] ostyakova.ln@gmail.com
Ксения Петухова[◇] petukhova.ka@mipt.ru
Вероника Смилга[◇] smilgaveronika@gmail.com
Диляра Жарикова[◇] dilyara.rimovna@gmail.com

[♡]Национальный исследовательский университет «Высшая школа экономики»
[◇]Московский физико-технический институт

Аннотация

Эта статья посвящена изучению иерархической и многоуровневой таксономии речевых функций. Чтобы оценить специфику близких прагматических классов, мы провели сравнительный анализ с участием как экспертов-аннотаторов, так и разметчиков краудсорсинга. Затем мы провели эксперименты по классификации аннотированного вручную набора данных и синтетического набора данных, сгенерированного с помощью ChatGPT. Мы рассмотрели возможность использования ChatGPT для получения данных для такой сложной сферы лингвистики, как дискурс. Данная работа вносит вклад в область лингвистической разметки данных.

Ключевые слова: речевые функции, ChatGPT, диалоговые системы, дискурс, общетематические диалоги

1 Introduction

The development of large language models (LLMs) such as ChatGPT, InstructGPT (Ouyang et al., 2022), DialoGPT (Zhang et al., 2019), GPT-3 (Brown et al., 2020), and others has contributed to the rapid expansion of Conversational AI. LLMs are often implemented in dialogue systems to generate replies to the user's utterances by using various prompt engineering techniques to elicit the required behaviour of the model. Incorporating LLMs makes conversational agents more adaptable, versatile, and simple to build. However, generative models need to be controlled within conversations with real users since they usually lack consistency, reliability, and common sense. Therefore, developers of conversational agents face a new challenge in light of the limitations of LLMs: the development of efficient methods to manage a dialogue flow.

Automatic discourse analysis is one of the most prominent ways of managing the dialogue flow in such systems because we can analyse and predict the structure of interconnected linguistic features: a topic, a speaker change, semantics, and pragmatics. For example, (Gu et al., 2021) present DialogBERT shifting the focus from utterance- to discourse-level in response generation. There are several fundamental theories for discourse analysis, such as Dialogue Act (DA) theory (Jurafsky et al., 1998), Segmented Discourse Representation Theory (SDRT) (Lascarides and Asher, 2007) , and Rhetorical Structure Theory (RST) (Mann and Thompson, 1987). Despite numerous applications to real-world problems, there is no standard approach to analysing discourse structures, particularly within open-domain dialogue systems. Despite the fact that discourse analysis is mostly oriented on pragmatics, tagsets usually reflect not pragmatic but grammar features of utterances (e.g., yes/no question, statement).

In this research paper, we focus on an alternative tagset developed by S.Eggins and D.Slade (Eggins and Slade, 2004) and explore its potential for use in dialogue systems. The taxonomy of speech functions is hierarchical and multilayered, including not only pragmatics but also turn-taking, feedback, and topic switching. Because the scheme includes classes with close pragmatics, we conducted additional research to determine whether it is possible to differentiate them for experts and crowdsourcing workers. Furthermore, we performed classification experiments on a manually annotated dataset as well as a synthetic dataset generated using ChatGPT. As a result, this paper contributes to the study of LLMs' prompt engineering techniques for linguistic annotation.

2 Discourse Analysis with Speech Function Theory

To get an idea of the structure of the dialogue and better manage the flow of the conversation, researchers often use an analysis of discourse structures. Such an analysis is used to represent dialogues at different linguistic levels, with a focus on pragmatics, i.e. functions of utterances or intentions of speakers. There are two common approaches to the research of discourse structures in the dialogues: Dialogue Act Theory (DA) (Jurafsky et al., 1998) and Segmented Discourse Representation theory (SDRT) (Lascarides and Asher, 2007). Within DA theory, each elementary discourse unit (EDU) is given a pragmatic characteristic, whereas SDRT, which is based on Rhetorical Structure Theory (Mann and Thompson, 1987), asserts a certain pragmatic class to a relation between two EDUs. The theory of dialogue acts is easier to apply to real-world problems since the task is carried out in one stage, unlike the SDRT approach, in which first the connections between statements must be determined and then only the connections are classified as discourse relations. For instance, a tagset of MIDAS, one of interpretations of DA theory, was used to select suitable replies in the Gunrock 2.0 chatbot, one of the participants in the Amazon Alexa Prize competition (Liang et al., 2020).

A number of tagsets were developed within DA theory and have gained prominence: DAMSL or Dialogue Act Markup in Several Layers (Core and Allen, 1997), Switchboard - DAMSL (Jurafsky, 1997), Meeting Recorder (Shriberg et al., 2004), and MIDAS (Yu and Yu, 2019). Interpretations differ in terms of discourse units, dialogue domains, and a number of described levels that results in inconsistent data (Table 1) although they usually have the same tags for general categories of utterances: statement, yes/no question, positive answer, negative answer.

Following SDTR, researches use one tagset in different task that inherits features of Rhetorical Structure Theory applied for text analysis. There are 16 labels for describing connections between utterances:

Clarification question, Comment, Question-answer pair etc (Li et al., 2020). However, existing datasets with such an annotation are task-oriented so they can not be used for analysis of casual conversations (see Table 1).

Theory	Dataset	Number of Utterances	Number of Labels	Domain
DA theory	SWITCHBOARD	205 000	60	open
	MRDA	180 000	54	open
SDRT theory	MOLWENI	88 303	16	technologies
	STAC	2 500	16	games

Table 1: Comparing of the most popular datasets with discourse annotation

Due to the lack of consistent conversational data with annotations that are good for open-domain dialogue systems, we decided to look into the potential of another taxonomy with classes similar to dialogue acts but with more functional dimensions for discourse analysis. It is important to mention that the theory of speech functions not only includes more complicated pragmatic categories than other taxonomies but also other layers of linguistic annotation that compound complicated discourse patterns united by a particular topic.

2.1 Speech Function Theory

(Eggins and Slade, 2004) developed a taxonomy of speech functions for discourse analysis of casual conversations extending M.K. Halliday's ideas about defining speakers' goals in dialogues. Speech functions combines features of DA theory and RST that reflects in connecting various layers of annotation in the system of dialogue turns and cross-dialogue discourse structure patterns (see Figure 1). Tagset developed by S.Eggins and D.Slade consists of speech functions representing different dimensions: Turn Management, Discourse Structure, Topic Organisation, Feedback (see Figure 1), Communicative Act, or Pragmatic Purpose.

Mostly, EDUs are defined by the functionality of dialogue acts within a particular theory used for discourse analysis. (Bunt et al., 2017) highlights the importance of defining EDUs by DA functions and even names units as functional segments. The speech function taxonomy differs from other approaches in terms of dialogue segmentation on EDUs as classes have more than one function. The taxonomy is divided into two levels of segmentation. The level of topics defines discourse patterns within conversations, while all speech functions are assigned at the sentence level. However, not all utterances are divided just into sentences; some of them are combined based on their common function or divided into several segments in other cases.

There are three high-level types of **discourse moves** in the taxonomy:

- Opening moves
- Sustaining moves
- Moves of Reaction

The purpose of **Opening moves** is to introduce new topics or start a conversation. According to S.Eggins and D.Slade, each Opening move indicates not only a new topic or the beginning of interaction between interlocutors within a conversation but also a **discourse pattern** (Eggins and Slade, 2004). **Sustaining moves** do not contribute to topic development but provide additional details and clarifications about the current topic given by the same speaker. They enhance the information discussed within it, while the speaker's role remains unchanged. **Moves of Reaction** are turns in dialogue where a speaker changes or responds to the previous utterance of the interlocutor that have more layers than the others. They are divided into two groups of speech functions representing different approaches to topic development. The React.Respond speech functions finish the conversation by not adding new challenges (e.g., questions changing conversational flow). React.Rejoinder, however, promotes discussion (see Appendix A).

Such a multilayered structure appears to be difficult to comprehend, especially given the uneven distribution of dimensions in tags. However, such complex dialogue modelling allows for the description of a conversational structure at various levels while taking into account topic shifts, discourse patterns, and

abstract intentions. The speech function annotation scheme, in contrast to other DA, SRDT taxonomies, has grammatical criteria for tag identification but does not include them in the tags. Besides that, speech functions feature a more subtle division into pragmatic classes comparing to other theories. For instance, most existing schemes for discourse analysis use the tag 'positive answer' for all cases when a speaker provides a yes-answer, while speech function theory distinguishes whether a speaker agrees with the provided information, acknowledges it, or affirms something.

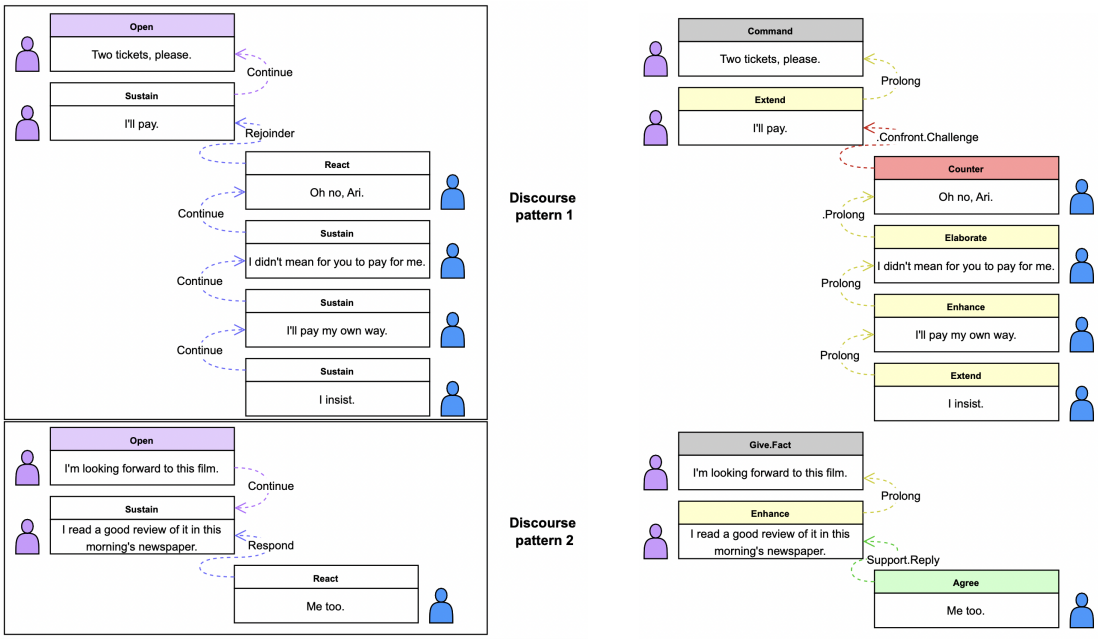

Figure 1: Discourse Patterns (left) and Feedback in Speech Functions (right)

3 Speech Function Dataset

Based on the classification developed in the framework of Speech Function Theory, we aim to obtain a dataset of open-domain dialogues with complex discourse annotation. The multidimensionality of the annotation scheme will allow to use the results in a variety of NLP tasks, especially those related to automatic discourse analysis.

As the basis for our speech function annotated dataset, we select DailyDialog, a dataset of human-written multi-turn dialogues on a variety of topics, widely used in evaluating open-domain dialogue systems. We preprocess DailyDialog data, removing duplicate dialogues and segmenting the remaining ones to split each utterance into several discourse units. To do so, we use a model for sentence segmentation that splits long and complex utterances into sentences and recovers punctuation.

As the first step of annotation, we employed three expert linguists to gather a small gold standard corpus with professionally annotated utterances. The resulting corpus consists of 75 dialogues (1264 utterances) annotated by three experts. We implemented an approach of double annotation with adjudication on our data, as it is commonly used for labelling discourse structures (Prasad et al., 2008; Webber et al., 2016; Zhou and Xue, 2015). We divided the dialogues into three equal parts, each annotated by two annotators independently. In cases of disagreement, the third expert not involved in annotating a particular part was responsible for adjudication and decided on final labels. The next step of the annotation process is crowd-sourcing annotation with the use of Toloka [1] crowdsourcing platform (Pavlichenko et al., 2021).

[1] https://toloka.ai/tolokers/

3.1 Inter-annotator Agreement: Experts vs. Crowdsourcing

(Mattar and Wachsmuth, 2012) implemented speech function annotation in a task-oriented dialogue system to aid in controlling a dialogue flow that demonstrated the possible potential of using the taxonomy for analyzing discourse structures. However, to work on automatic analysis using speech functions in open-domain dialogue systems, it was necessary to prove that the chosen taxonomy is reliable enough. So, we conducted several experiments on the annotation of casual conversations in English.

Annotation of discourse structures or dialogue acts is not trivial because it requires linguistic knowledge or trained workers (Yung et al., 2019). Besides that, perception of speakers' intentions in utterances differs across individuals, making the task even more difficult. We compared two results of annotation with speech functions completed by experts with professional backgrounds in linguistics and crowdsourced workers. We used Fleiss' Kappa (Fleiss and Cohen, 1973) for measuring inter-annotation agreement as it is considered to be the most common way to evaluate taxonomy reliability in tasks related to discourse analysis. However, this evaluation method has the limitation of not considering the common mistakes of annotators. That is why we measured not only inter-annotator agreement but also accuracy, weighted recall, and precision, as well as macro and micro F1 (Ghamrawi and McCallum, 2005), by comparing workers' annotations to results by experts.

Crowdsourcing is not the best option for labelling data with discourse structures since it is not possible to obtain high-quality annotations with linguistic labels from untrained workers (Kawahara et al., 2014). Nevertheless, it is important to test to what extent classes can be defined by non-professionals. For obtaining better results by crowdsourcing workers, we developed hierarchical guidelines consisting of easy questions about a topic and speaker change, the type of a sentence, the pragmatics of the utterance, and examples that allow better orientation in the scheme for untrained annotators (see Appendix B). In addition, extra methods for controlling the quality of annotation were devised to help us identify unreliable annotators, and some hints were included for crowdsourcing workers.

As a result of crowdsourcing, 675 utterances were cross-annotated by three non-professional workers each. It is important to note that crowdsourcing workers were different in each case that could also cause inconsistency. We evaluated the results for 16 high-level cut labels and the complete taxonomy to determine the weak points of the established hierarchical guidelines. Cut labels group the classes that are really close to each other in terms of pragmatics into one class (see Appendix B). When measuring the quality of crowdsourced annotation, we also examined cases of voting where not all annotators but the majority agree on a tag (see Table 2). As for cut labels, they were labeled with pretty good accuracy by crowdsourcing workers. Annotation of full tags is more challenging for non-experts, which is proven by all metrics. Macro F1 value shows that we have to pay attention to improving quality of annotating low-level classes (see Table 2). Measuring inter-annotator agreement using Fleiss' Kappa proves that the tags with close pragmatics are difficult for differentiating not only for untrained workers, but for experts as well. Still, in case of experts' annotation, Fleiss' kappa is more than 0.6, meaning that the chosen taxonomy is quite reliable (see Figure 5).

To sum up, crowdsourcing is a very consuming process in terms of time and resources, especially for such complicated annotation tasks related to linguistic data augmentation. Furthermore, this method of enlarging labeled data is not so effective as values of accuracy metrics and Fleiss' kappa have shown. The data labeled by crowdsourcing workers needs to be corrected by experts, which slows down and complicates the annotation process. That is why our next experiments on data augmentation were conducted using large language models.

3.2 Generating a Synthetic Speech Functions Dataset with ChatGPT

Data augmentation is a technique widely used in machine learning to increase the size of the training data. It can be especially useful when dealing with limited or imbalanced data, improving generalization and preventing overfitting. (Wei and Zou, 2019) describes a set of simple data augmentation methods that significantly improve the performance of models such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs) on text classification tasks. In (Kobayashi, 2018), the authors

	Accuracy	Weighted Recall	Weighted Precision	Macro F1	Micro F1
Full tags	0.52	0.52	0.62	0.37	0.55
Full tags + voting	0.54	0.54	0.62	0.37	0.54
Cut labels	0.83	0.83	0.83	0.53	0.83
Cut labels + voting	0.87	0.87	0.85	0.53	0.87

Table 2: Evaluation of annotation by crowdsourcing workers

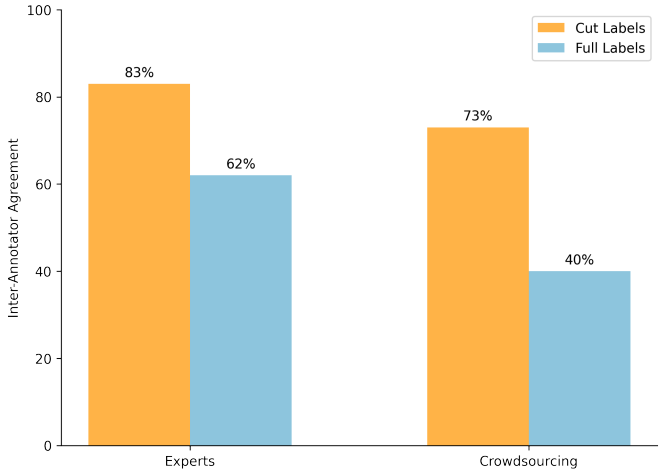

Figure 2: Inter-annotator Agreement

pretrain an LSTM on Wikipedia articles and fine-tune it on several labelled datasets to generate more sentences from training data by using the fine-tuned model to replace some words. Again, the proposed method improved the RNNs and CNNs performance on text classification tasks. In (Xie et al., 2020), the authors explore various advanced methods of data augmentation for language and vision tasks. On IMDb text classification dataset, their model trained on only 20 labelled examples mixed with augmented data outperforms the original state-of-the-art model trained on approximately 25000 labelled examples. Finally, (Kumar et al., 2020) describes how pre-trained text generation models like BART, BERT and GPT-2 can be used to generate augmented text data.

As we are now in the beginning of the process of building a speech functions dataset and lack annotated data, we decided to test whether we could effectively use data augmentation methods to build a decently performing classification model. In addition to that, any speech function dataset is by its nature imbalanced, as some speech functions are seen many times more rarely in conversations than the others, which would also make data augmentation methods effective. ChatGPT is a pretrained generative text model which was fine-tuned using reinforcement learning with human-feedback data. As reported in (OpenAI, 2022) and (Ouyang et al., 2022), InstructGPT and its sibling model ChatGPT perform particularly well when given instructions in natural language. Following (Kumar et al., 2020) and (Kobayashi, 2018) who use language models for textual data augmentation, we decided to use ChatGPT to generate synthetic data for our speech functions dataset.

The model was accessed via OpenAI API [2] and provided with hand-crafted instructions for each speech function class. We tried to implement different strategies in order to get more suitable, natural and various conversational data for particular classes:

- to make the model follow instructions developed for crowdsourcing and label the whole dialogue;

[2] `https://platform.openai.com/overview`

- to give instructions only with description of classes;
- to give just examples of classes;
- to give examples of one speech function;
- to give examples of several similar classes.

We had a lot of challenges putting the above-mentioned data generation strategies into action because of ChatGPT's limitations. The model overuses certain phrases that interfere with generating various conversational data. Even mentioning a change of topic and word collocations in prompts does not always lead to the variety of results needed. The instability of generative models does not allow to generate similar data with the same instructions. So, working on data augmentation, we had to control such cases of unstable generation and remove them from the data. As we were working with linguistic annotation, the model interpreted some labels differently than they were given in the instruction.

Considering all experiments, the final instruction included 1) the speech function name; 2) the speech function definition; 3) examples from the expert-annotated Gold Standard dataset; 4) guidelines for the model, i.e. "Generate 20 datapoints from these examples" (see Appendix C for a prompt example). We generated from 500 to 1000 datapoints for each class, approximately 25000 speech function examples in total (see AppendixD). We also generated examples to train a separate classification model to distinguish between declarative, interrogative, and miscellaneous (that includes emotional exclamations, greetings, goodbyes, etc.) classes.

4 Classification

We developed a multi-level annotation pipeline (see Figure 3) to annotate dialogues with Speech Functions. Firstly, a Topic Shift Classifier is applied to determine if an utterance initiates a new topic. Subsequently, an Upper Level Classifier annotates all utterances by identifying the type of the utterance. If the utterance is interrogative, the question classifier is then used to obtain the final label. If the utterance is declarative or miscellaneous, the Declarative Classifier or Miscellaneous Classifier is used, respectively. For utterances that were defined as commands, the final label is also 'COMMAND'. Definitions and examples of all final labels can be found in Table 5 of the Appendix.

The DeepPavlov library (Burtsev et al., 2018) was used to train classifiers for our project. For the Topic Shift Classifier, we trained double sequence binary classifier model based on `roberta-large-mnli`, where the input was a sequence of two consecutive utterances. The true label denotes the topic shift in the utterances. The model was trained with the following hyper-parameters: learning rate – 2e-5, optimizer – AdamW, input max length – 128. We applied the early-stopping to successfully train the model. Using pre-trained model allowed the classifier to transfer knowledge gained while pre-training on mnli to related task of shift identification (Konovalov et al., 2020; Gulyaev et al., 2020).

Similarly, for our remaining classifiers, we utilized double sequence classification based on `bert-base-cased` multi-class classification.

Table 3 shows the evaluation results on the test set of ChatGPT data. Table 4 displays the evaluation results for real data, i.e., dialogues that were manually annotated by the experts.

Classifier	**Accuracy**
Topic Shift	0.86
Upper Level	0.99
Questions	0.97
Declarative	0.94
Miscellaneous	0.99

Table 3: Evaluation results on ChatGPT data

Overall, it is evident that the accuracy of all classifiers, except the Topic Shift Classifier, is significantly lower on real data. The low level of classification quality for declarative and interrogative utterances can be explained by two main reasons. Firstly, distinguishing between Speech Functions within interrogative and declarative classes is challenging, even for humans, as shown in Table 2. Secondly, the data samples

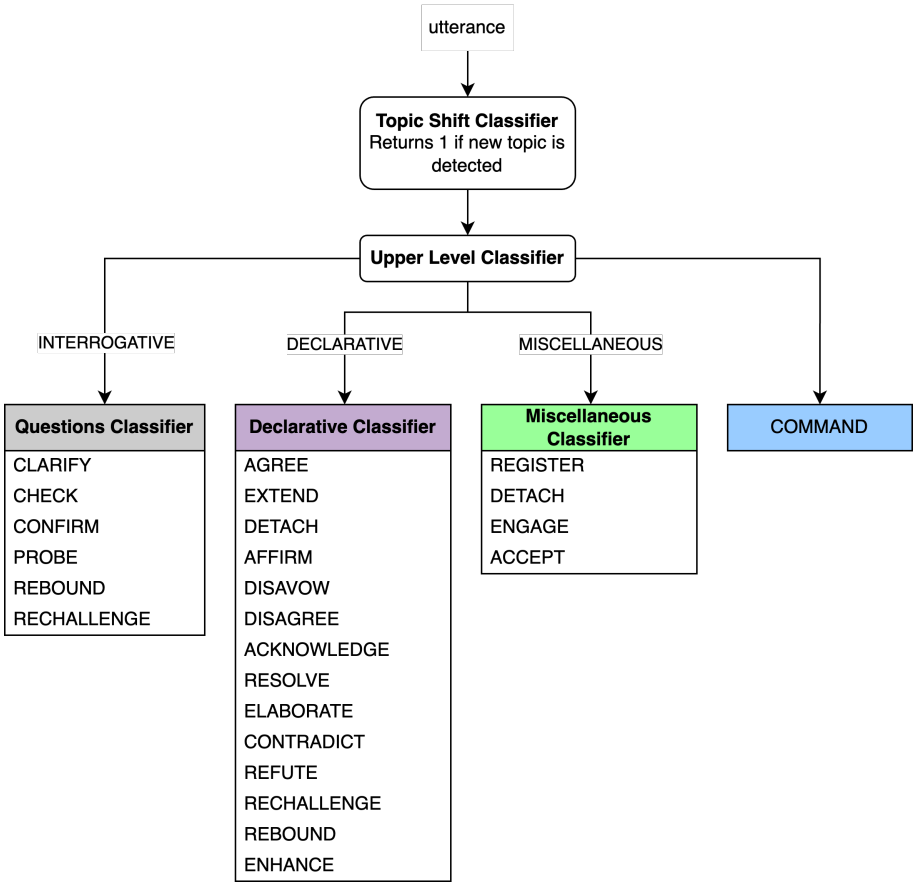

Figure 3: Annotation pipeline

Classifier	Accuracy	Weighted Recall	Weighted Precision	Weighted F1
Topic Shift	0.91	0.91	0.96	0.93
Upper Level	0.60	0.60	0.87	0.71
Questions	0.34	0.34	0.83	0.43
Declarative	0.20	0.20	0.31	0.24
Miscellaneous	0.81	0.81	0.87	0.84
Random Topic Shift	0.53	0.53	0.57	0.55
Random Upper Level	0.25	0.25	0.55	0.34
Random Questions	0.20	0.20	0.61	0.27
Random Declarative	0.09	0.09	0.20	0.15
Random Miscellaneous	0.23	0.23	0.55	0.33

Table 4: Evaluation results on real dialogues

generated with ChatGPT are very similar within classes. Although different prompts and examples were used during the generation process, samples are syntactically and semantically alike. Consequently, the model learned to differentiate between highly specific and similar samples of Speech Functions, while real conversations are much more unpredictable and varied, making it harder for the model to accurately classify them. Thus, for prompt illustrated in Figure 4, ChatGPT generated several similar examples on cuisine topic.

Here are some of them:

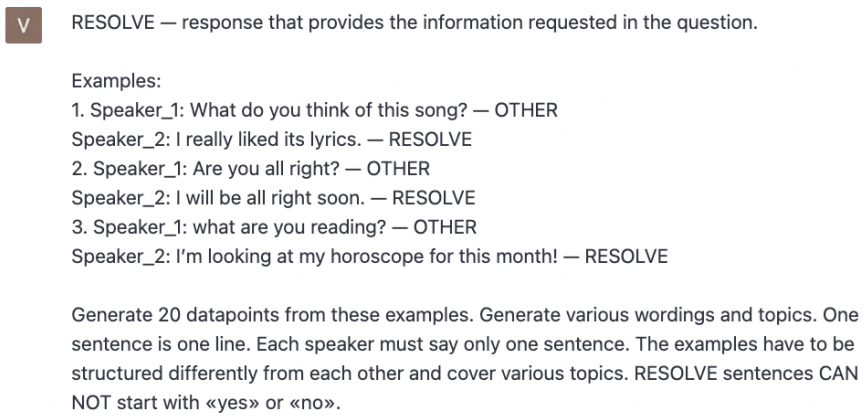

Figure 4: Prompt for generation of RESOLVE samples

- Speaker_1: What's your favorite type of cuisine? — OTHER
  Speaker_2: I love Mexican food, especially tacos! — RESOLVE
- Speaker_1: What's your favorite food? — OTHER
  Speaker_2: I love sushi and could eat it every day! — RESOLVE
- Speaker_1: What's your favorite type of cuisine? — OTHER
  Speaker_2: I love Japanese food, especially sushi and ramen. — RESOLVE

5 Conclusion and Future Work

This paper gives a thorough look at research done on a new way to analyze discourse in open-domain dialogue systems. Speech function theory sees the discourse structure of dialogues as a complex hierarchical system that connects linguistic levels and functional dimensions like taking turns, changing topics, pragmatics, and the interlocutor's feedback. Of particular research interest was the fact that low-levels of speech functions all reflect pragmatics, not semantics, as in many popular taxonomies. We checked the reliability of the taxonomy and did experiments on labelling dialogues on casual topics from the DailyDialog dataset, comparing inter-annotator agreement between experts with backgrounds in linguistics and untrained crowdsourcing workers. Considering the results of experts' annotation, it was proven that the scheme for annotation is reliable enough but still difficult because of close classes in terms of pragmatics.

In our study, we employed ChatGPT to generate synthetic data for our speech functions dataset as the human-labelled dataset is imbalanced which makes training a classifier more difficult. While exploring ChatGPT's capabilities, we found several strategies to create suitable conversational data for each speech function class. We encountered several challenges due to the nature of language models, such as overuse of certain phrases and instability in generation. However, by refining our instructions and incorporating expert-annotated examples from the Gold Standard dataset, we managed to generate 27,000 datapoints. Based on the generated data, we trained a custom multi-level annotation pipeline. The pipeline includes a Topic Shift Classifier, an Upper Level Classifier, a Question Classifier, a Declarative Classifier, and a Miscellaneous Classifier. The results show that the accuracy of the classifiers is significantly lower on real data, which can be attributed to the challenges of distinguishing between Speech Functions within interrogative and declarative classes and the limited variability of the data generated by ChatGPT.

Our next steps will involve running experiments on classification with ChatGPT because we could not achieve satisfactory results for speech function classification using data generation as an augmentation method. As LLMs pre-trained on instructions are becoming more popular instruments for data augmentation, implementing other models for labelling or generation may be beneficial to our research. In order to improve metrics for this classification task, we also intend to try training or fine-tuning other Transformer models on the annotated dialogues.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Harry Bunt, Volha Petukhova, and Alex Chengyu Fang. 2017. Revisiting the iso standard for dialogue act annotation. // *Proceedings of the 13th Joint ISO-ACL Workshop on Interoperable Semantic Annotation (ISA-13)*.

Mikhail Burtsev, Alexander Seliverstov, Rafael Airapetyan, Mikhail Arkhipov, Dilyara Baymurzina, Nickolay Bushkov, Olga Gureenkova, Taras Khakhulin, Yuri Kuratov, Denis Kuznetsov, and Vasily Konovalov. 2018. Deeppavlov: Open-source library for dialogue systems. // *NIPS*.

Mark G Core and James Allen. 1997. Coding dialogs with the damsl annotation scheme. // *AAAI fall symposium on communicative action in humans and machines*, volume 56, P 28–35. Boston, MA.

Suzanne Eggins and Diana Slade. 2004. *Analysing casual conversation*. Equinox Publishing Ltd.

Joseph L Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Nadia Ghamrawi and Andrew McCallum. 2005. Collective multi-label classification. // *Proceedings of the 14th ACM international conference on Information and knowledge management*, P 195–200.

Xiaodong Gu, Kang Min Yoo, and Jung-Woo Ha. 2021. Dialogbert: Discourse-aware response generation via learning to recover and rank utterances. // *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, P 12911–12919.

Pavel Gulyaev, Eugenia Elistratova, Vasily Konovalov, Yuri Kuratov, Leonid Pugachev, and Mikhail Burtsev. 2020. Goal-oriented multi-task bert-based dialogue state tracker.

Dan Jurafsky, Elizabeth Shriberg, Barbara Fox, and Traci Curl. 1998. Lexical, prosodic, and syntactic cues for dialog acts. // *Discourse Relations and Discourse Markers*.

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*.

Daisuke Kawahara, Yuichiro Machida, Tomohide Shibata, Sadao Kurohashi, Hayato Kobayashi, and Manabu Sassano. 2014. Rapid development of a corpus with discourse annotations using two-stage crowdsourcing. // *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, P 269–278.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201*.

Vasily Konovalov, Pavel Gulyaev, Alexey Sorokin, Yury Kuratov, and Mikhail Burtsev. 2020. Exploring the bert cross-lingual transfer for reading comprehension. // *Komp'juternaja Lingvistika i Intellektual'nye Tehnologiithis*, P 445–453.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. *arXiv preprint arXiv:2003.02245*.

Alex Lascarides and Nicholas Asher. 2007. Segmented discourse representation theory: Dynamic semantics with discourse structure. *Computing meaning*, P 87–124.

Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, and Bing Qin. 2020. Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure. *arXiv preprint arXiv:2004.05080*.

Kaihui Liang, Austin Chau, Yu Li, Xueyuan Lu, Dian Yu, Mingyang Zhou, Ishan Jain, Sam Davidson, Josh Arnold, Minh Nguyen, et al. 2020. Gunrock 2.0: A user adaptive social conversational system. *arXiv preprint arXiv:2011.08906*.

William C Mann and Sandra A Thompson. 1987. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute Los Angeles.

Nikita Mattar and Ipke Wachsmuth. 2012. Small talk is more than chit-chat: Exploiting structures of casual conversations for a virtual agent. *// KI 2012: Advances in Artificial Intelligence: 35th Annual German Conference on AI, Saarbrücken, Germany, September 24-27, 2012. Proceedings 35*, P 119–130. Springer.

OpenAI. 2022. Introducing chatgpt. *OpenAI Blog*. Accessed: 2023-03-17.

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.

Nikita Pavlichenko, Ivan Stelmakh, and Dmitry Ustalov. 2021. Crowdspeech and voxdiy: Benchmark datasets for crowdsourced audio transcription. *arXiv preprint arXiv:2107.01091*.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. 2008. The penn discourse treebank 2.0. *// LREC*.

Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang, and Hannah Carvey. 2004. The icsi meeting recorder dialog act (mrda) corpus. Technical report, INTERNATIONAL COMPUTER SCIENCE INST BERKELEY CA.

Bonnie Webber, Rashmi Prasad, Alan Lee, and Aravind Joshi. 2016. A discourse-annotated corpus of conjoined vps. *// Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, P 22–31.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268.

Dian Yu and Zhou Yu. 2019. Midas: A dialog act annotation scheme for open domain human machine spoken conversations. *arXiv preprint arXiv:1908.10023*.

Frances Yung, Vera Demberg, and Merel Scholman. 2019. Crowdsourcing discourse relation annotations by a two-step connective insertion task. *// Proceedings of the 13th Linguistic Annotation Workshop*, P 16–25.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.

Yuping Zhou and Nianwen Xue. 2015. The chinese discourse treebank: A chinese corpus annotated with discourse relations. *Language Resources and Evaluation*, 49:397–431.

A Speech Functions list

Speech Function	Definition
Open.Attend	These are usually greetings. NB: Used in the beginning of a conversation. Example: Hi!
Open.Demand	Demanding information. NB: Used in the beginning of a conversation. Example: What's Allenby doing these days?
Open.Give	Providing information. NB: Used in the beginning of a conversation. Example: I met his sister.
Open.Command	Making a request, an invitation or command to start a dialogue or discussion of a new topic. Example: Let's go for a walk!
Sustain.Continue.Prolong.**Extend**	Adding supplementary or contradictory information to the previous statement. A declarative sentence or phrase (may include and, but, except, on the other hand). Example: Just making sure you don't miss the boat. **I put it out on Monday mornings. I hear them. I hate trucks.**
Sustain.Continue.Prolong.**Elaborate**	Clarifying / rephrasing the previous statement or giving examples to it. A declarative sentence or phrase (may include for example, I mean, like). Example: Yeah but I don't like people... um... **I don't want to be INVOLVED with people.**
Sustain.Continue.Prolong.**Enhance**	Adding details to the previous statement, adding information about time, place, reason, etc. A declarative sentence or phrase (may include then, so, because). Example: Nor for much longer. **We're too messy for him.**
Sustain.Continue.**Monitor**	Checking the involvement of the listener or trying to pass on the role of speaker to them. Example: You met his sister that night we were doing the cutting and pasting up. **Do you remember?**
React.Rejoinder.Confront.Response.**Re-challenge**	Offering an alternative position, often an interrogative sentence. Example: David: Messi is the best. Nick: **Maybe Pele is the best one?**
React.Rejoinder.Support.Challenge.**Rebound**	Questioning the relevance, reliability of the previous statement, most often an interrogative sentence. Example: David: This conversation needs Allenby. Fay: **Oh he's in London.** So what can we do?
React.Rejoinder.Support.Response.**Resolve**	The response provides the information requested in the question. Example: Lina: What do you think of this song? Fay: **I really like its lyrics.**
React.Rejoinder.Support.Track.**Check**	Getting the previous speaker to repeat an element or the entire statement that the speaker has not heard or understood. Example: Straight into the what?
React.Rejoinder.Support.Track.**Clarify**	Asking a question to get additional information on the current topic of the conversation. Requesting to clarify the information already mentioned in the dialogue. Example: What, before bridge?

React.Rejoinder.Support.Track.**Confirm**	Asking for a confirmation of the information received. Example: David: Well, he rang Roman, he rang Roman a week ago. Nick: **Did he?**
React.Rejoinder.Support.Track.**Probe**	Requesting a confirmation of the information necessary to make clear the previous speaker's statement. The speaker themselves speculates about the information that they want to be confirmed. Example: Because Roman lives in Denning Road also?
React.Respond.Confront.Reply.**Contradict**	Refuting previous information. No, sentence with opposite polarity. If the previous sentence is negative, then this sentence is positive, and vice versa. NB! The speaker contradicts the information that he already knew before. Example: Fay: Suppose he gives you a hard time, Nick? Nick: **Oh I like David a lot.**
React.Respond.Confront.Reply.**Disagree**	Negative answer to a question or denial of a statement. No, negative sentence. Example: Fay: David always makes a mess in our room. May: **No, he's not so bad.**
React.Respond.Confront.Reply.**Disavow**	Denial of knowledge or understanding of information. Example: I don't know.
React.Respond.Support.Develop.**Elaborate**	Clarifying / rephrasing the previous statement or giving examples to it. A declarative sentence or phrase (may include for example, I mean, like). Example: Nick: Cause all you'd get is him bloody raving on. Fay: **He's a bridge player, a naughty bridge player.**
React.Respond.Support.Develop.**Enhance**	Adding details to the previous statement, adding information about time, place, reason, etc. A declarative sentence or phrase (may include then, so, because). Example: Fay: He kept telling me that. Nick: **The trouble with Roman though is that — you know he does still like cleaning up.**
React.Respond.Support.Develop.**Extend**	Adding supplementary or contradictory information to the previous statement. A declarative sentence or phrase (may include and, but, except, on the other hand). Extend: David: That's what the cleaner — your cleaner lady cleaned my place thought. Nick: **She won't come back to our place.**
React.Respond.Support.**Engage**	Drawing attention or a response to a greeting. Example: Hey, David.
React.Respond.Support.**Register**	A manifestation of emotions or a display of attention to the interlocutor. Example: Yeah.
React.Respond.Support.Reply.**Acknowledge**	Indicating knowledge or understanding of the information provided. Example: I know.
React.Respond.Support.Reply.**Affirm**	A positive answer to a question or confirmation of the information provided. Yes/its synonyms or affirmation. NB! The speaker confirms the information that he already knew before. Example: Nick: He went to London. Fay: **He did.**

React.Respond.Support.Reply.**Accept**	Expressing gratitude. Example: Thank you!
React.Respond.Support.Reply.**Agree**	Agreement with the information provided. In most cases, the information that the speaker agrees with is new to him. Yes/its synonyms or affirmation. Example: Steve: We're gonna make it. Mike: **Yeah, right.**

Table 5: Speech functions and their communicative roles in the dialogue

B Cut and full Speech Function labels

Cut labels	**Full labels**
Open.Demand	Open.Demand
Open.Give	Open.Give
Open.Command	Open.Command
Open.Attend	Open.Attend
React.Rejoinder.Confront.Response	React.Rejoinder.Confront.Response.Re-challenge
React.Rejoinder.Support.Track	React.Rejoinder.Support.Track.Probe
	React.Rejoinder.Support.Track.Check
	React.Rejoinder.Support.Track.Clarify
	React.Rejoinder.Support.Track.Confirm
Sustain.Continue. Prolong	Sustain.Continue. Prolong.Extend
	Sustain.Continue. Prolong.Enhance
	Sustain.Continue. Prolong.Elaborate
React.Rejoinder.Support.Challenge.Rebound	React.Rejoinder.Support.Challenge.Rebound
React.Respond.Support.Reply	React.Respond.Support.Reply.Affirm
	React.Respond.Support.Reply.Acknowledge
	React.Respond.Support.Reply.Agree
React.Respond.Support.Develop	React.Respond.Support.Develop.Extend
	React.Respond.Support.Develop.Enhance
	React.Respond.Support.Develop.Elaborate
React.Respond.Confront.Reply	React.Respond.Confront.Reply.Disagree
	React.Respond.Confront.Reply.Contradict
	React.Respond.Confront.Reply.Disavow
Sustain.Continue.Monitor	Sustain.Continue.Monitor
React.Respond.Support.Register	React.Respond.Support.Register
React.Respond.Support.Engage	React.Respond.Support.Engage
React.Respond.Support.Accept	React.Respond.Support.Accept
React.Rejoinder.Support.Response.Resolve	React.Rejoinder.Support.Response.Resolve

C Annotation interface and prompt example

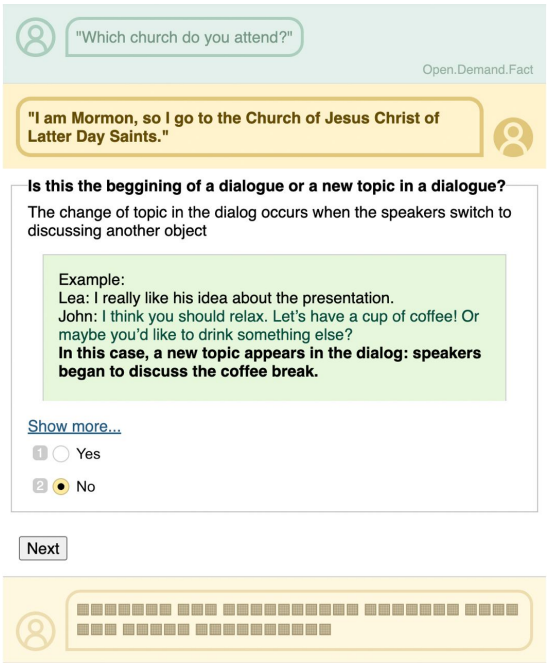

Figure 5: Guidelines for annotators

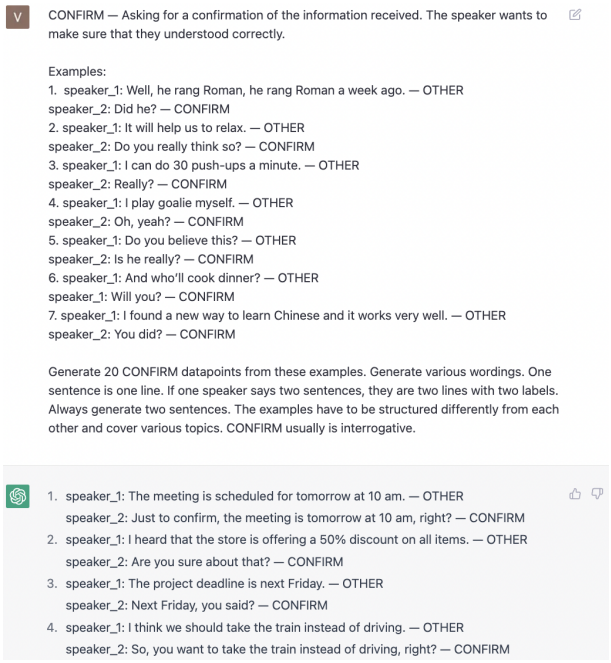

Figure 6: An example of a prompt and generation results

D Dataset statistics

	Class name	Synthetic dataset	Original dataset
Upper level	INTERROGATIVE	790	
	DECLARATIVE	738	
	MISCELLANEOUS	655	
	COMMAND	630	14
Declarative classes	AGREE	980	49
	EXTEND	996	383
	AFFIRM	933	54
	DISAVOW	800	7
	DISAGREE	774	39
	ACKNOWLEDGE	688	9
	RESOLVE	583	103
	ELABORATE	571	91
	CONTRADICT	544	2
	REFUTE	588	-
	RECHALLENGE	530	2
	REBOUND	511	5
	ENHANCE	424	77
Miscellaneous classes	REGISTER	502	78
	DETACH	630	4
	ENGAGE	504	6
	ACCEPT	307	17
Interrogative classes	CLARIFY	564	162
	CHECK	665	14
	CONFIRM	591	23
	PROBE	574	39
	REBOUND	543	5
	RECHALLENGE	509	2

E Metrics

- **The Fleiss' kappa** statistic is used to examine the level of agreement among multiple assessors evaluating a categorical or nominal variable. It is calculated by comparing observed and expected agreement among raters. The range of Fleiss' kappa is 0 to 1 where 1 implies full agreement. A value of 0.6 or more is considered to be a good agreement.

$$Fleiss' kappa = \frac{P_o - P_e}{1 - P_e}$$

  - P_o is the observed agreement among the raters;
  - P_e is the expected agreement by chance, which is calculated based on the marginal frequencies of the categories being rated.
- **Accuracy** measures how accurately a model or classifier predicts the proper outcome or label for a dataset. The model or classifier's accuracy score is the percentage of correct predictions.

$$accuracy = \frac{number\ of\ correct\ predictions}{total\ number\ of\ predictions}$$

- **Weighted Recall and Precision** in classification tasks where the classes are imbalanced. Recall is an evaluation of a model's capacity to identify all relevant instances of a target class. Precision is a measure of a model's ability to identify only instances of a target class that are relevant. In both cases, the weights wi can be adjusted to the proportion of instances in each class or to a value based on class importance.

$$weighted\ precision = \frac{\sum_{i=1}^{N} w_i \cdot TP_i}{\sum_{i=1}^{N} w_i \cdot (TP_i + FP_i)}$$

  - N is the number of classes;
  - w_i is the weight of class i;
  - TP_i is the number of true positives for class i;
  - FP_i is the number of false positives for class i.

$$weighted\ recall = \frac{\sum_{i=1}^{N} w_i \cdot TP_i}{\sum_{i=1}^{N} w_i \cdot (TP_i + FN_i)}$$

  - N is the number of classes;
  - w_i is the weight of class i;
  - TP_i is the number of true positives for class i;
  - FN_i is the number of false negatives for class i.
- **Micro F1** is a dataset-wide F1 score. Precision, recall, and F1 scores are obtained by measuring the total number of true positives, false positives, and false negatives across all classes.

$$F_1^{micro} = \frac{2 \cdot TP_{total}}{2 \cdot TP_{total} + FP_{total} + FN_{total}}$$

  - TP_{total} is the total number of true positives;
  - FP_{total} is the total number of false positives;
  - FN_{total} is the total number of false negatives across all classes.
- **Macro F1** is calculated for each class and averaged. It weights each class equally regardless of dataset frequency.

$$F_1^{macro} = \frac{1}{N} \sum_{i=1}^{N} F_1^i$$

  - N is the number of classes;
  - F_1^i is the F1 score for class i.