Speaker Clustering in Textual Dialogue with Pairwise Utterance Relation and Cross-corpus Dialogue Act Supervision

Zhihua Su[1] and **Qiang Zhou**[1,2,*]
[1]State Key Laboratory of Intelligent Technology and Systems,
Department of Computer Science and Technology, Tsinghua University
[2]Beijing National Research Center for Information Science and Technology, Tsinghua University
suzh20@mails.tsinghua.edu.cn
zq-lxd@mail.tsinghua.edu.cn

Abstract

We propose a speaker clustering model for textual dialogues, which groups the utterances of a multi-party dialogue without speaker annotations, so that the actual speakers are identical inside each cluster. We find that, without knowing the speakers, the interactions between utterances are still implied in the text, which suggest the relations between speakers. In this work, we model the semantic content of utterance with a pre-trained language model, and the relations between speakers with an utterance-level pairwise matrix. The semantic content representation can be further instructed by cross-corpus dialogue act modeling. The speaker labels are finally generated by spectral clustering. Experiments show that our model outperforms the sequence classification baseline, and benefits from the auxiliary dialogue act classification task. We also discuss the detail of determining the number of speakers (clusters), eliminating the interference caused by semantic similarity, and the impact of utterance distance.

1 Introduction

Processing dialogues is a classical linguistic task. With the development of pre-trained language models in recent years, studies on dialogues have made great progress (Zhang et al., 2020; Roller et al., 2021; Adiwardana et al., 2020). In general, these training processes, especially pre-training, need a large amount of data. Meanwhile, most of dialogue models are designed to input speaker information, for example, applying trainable speaker embeddings, or just assuming the dialogue is composed of two speakers involved turn by turn, to introduce dialogue structure information into the models. But for common researchers, dialogue data is hard to collect. Datasets like subtitles (Lison et al., 2018) contain a lot of dialogue data of daily communication, but lack of speaker annotation. Some researches in related fields, such as conference transcription (Raj et al., 2021; Fu et al., 2021; Kanda et al., 2022) and multimodal body tracking (Vallet et al., 2016; Nickel et al., 2005; Wang and Brandstein, 1999), may also be improved by text-based speaker clustering techniques. Speaker clustering can also be a self-supervision dialogue pre-training procedure in the scenario that speaker annotation is adequate. Therefore, it is valuable to develop a model to reconstruct the missing identities of speakers in textual dialogue data.

In order to reconstruct the speaker labels in the dialogue, this work is dedicated to the method of speaker clustering. Different from previous researches on speaker identification (Kundu et al., 2012; Ma et al., 2017; Ek et al., 2018), which aim at selecting the most similar speaker from the pre-modeled candidates, the speaker clustering task aims at grouping the utterances into speaker-specific clusters without any preset candidates (Lukic et al., 2016). It is more useful because it works on open corpus where the speakers cannot be modeled in advance.

Speaker clustering is relevant to dialogue structure, because the process of turns follows certain patterns. These patterns include the semantic content and the communicative functions of utterance, and can be specifically represented as the dialogue act (DA) of utterance and associations between dialogue acts respectively (Bunt et al., 2010). The associations between dialogue acts include question-answer, request-response, offer-acceptance, etc., which are closely related to alternation of speakers. Conversely, the relations between speakers will be predicable if these patterns are available from textual utterances.

In this work, the speaker relations are inferred from the communicative functions by using an utterance-level pairwise matrix. The speaker relations have only two possible values, either same or different. The relations among the whole dia-

*Corresponding author.

Proceedings of the 29th International Conference on Computational Linguistics, pages 734–744
October 12–17, 2022.

logue form this matrix, which is regarded as the similarity matrix of the ground speakers.

The matrix can reconstruct the clusters of speakers with a density-based clustering method. The most popular algorithms of density-based clustering are spectral clustering (Von Luxburg, 2007) and DBSCAN (Hess et al., 2019). In this work, we use spectral clustering as the implementation, because it is less sensitive to sparse points, which follows this task that each utterance must be in a cluster.

Based on the above analysis, we build a model that models the semantic content of utterances with multi-task cross-corpus DA supervision, calculates the speaker relations with the form of bilinear, and generates the cluster labels with the method of spectral clustering.

The main contributions of this paper are summarized as follows.

- We build a speaker clustering model for textual dialogue, which explicitly exploits the communicative functions to reconstruct speaker relations and outperforms the baseline.
- The model can be further improved by auxiliary DA classification task. Even if a dataset is lack of DA annotations, the model can still be improved by cross-corpus DA data.
- We discuss the reliability of our method to predict the number of clusters, the ability to disambiguate between speaker relation and utterance text similarity, and the impact of utterance distance.

2 Related Work

This work targets for speaker clustering, and is based on the theories of dialogue structure.

2.1 Speaker Clustering

As far as we know, there are few works directly on speaker clustering in textual dialogues. However, there are some previous works on speaker diarization in voice conversations, and speaker clustering is the most important step in speaker diarization (Tranter and Reynolds, 2006; Anguera et al., 2012; Park et al., 2022). But these works only use audio features as the basis to calculate relation without considering the semantic information.

A previous work on speaker diarization through pairwise relations based on audio (Lin et al., 2019) uses spectral clustering as the top-level structure, which provides an idea for our structural design. But its focus is only audio features too, and it just descends the loss similarity score without training more fundamental features into fixed classes, which makes it difficult for the feature extraction process to guarantee generalization.

2.2 Dialogue Structure

The early researches in dialogue processing have noticed that a dialogue is made up of turns. Each turn is a combination of a speaker and an utterance. The turns are push ahead following the semantic cue. Specifically, dialogue turns have semantic content and communicative functions, which can be represented as dialogue acts (Searle and Searle, 1969) and adjacency pairs (Schegloff and Sacks, 1973) respectively. Every turn has its own dialogue act. Two turns from different speakers will form an adjacency pair if they have a behavior of interaction. Base on statistical or machine learning methods, it is realizable to predict the dialogue acts or the adjacency pairs (Surendran and Levow, 2006; Li et al., 2019; Li and Wu, 2016; Zhang et al., 2018). The semantic content and communicative functions involve the relations between speakers.

Pre-trained language models (Devlin et al., 2019; Lewis et al., 2020; Brown et al., 2020) have demonstrated their effectiveness on semantic modeling. These works illustrate the idea of represent semantic content with contextualized embeddings, i.e., trainable distributed vector in semantic space. However, most of the above models output word-level embeddings to represent the meaning of a word instead of the meaning of a whole sentence. There are solutions to convert from word-level embeddings to utterance-level embeddings, including using the corresponding embedding of the [CLS] token and using some pooling strategies (Ma et al., 2019; Xiao, 2018).

2.3 Other Works Related to Speakers in Dialogue

There are some researches relevant to speaker labeling in textual dialogues (Kundu et al., 2012; Ma et al., 2017; Ek et al., 2018), but they are not speaker clustering models directly. Most of them depend on the assumption that each speaker has its own speaking feature, e.g. the proportion of stop words, short words, adverbs in its utterances. Turn-taking detection is another type of speakers labeling (Liang and Zhou, 2020; Aldeneh et al.,

2018). It refers to identifying the positions where the speakers change during the dialogue, but it only focuses on the relations between two adjacent utterances, instead of every pair of utterances among a multi-party dialogue.

3 Model

The three main processes of this model are getting representation of utterances, cooperating with cross-corpus DA supervision, and calculating the similarity score. Therefore, as the overall structure shown in Figure 1, the model is divided into three parts in general: the utterance embedding part (blue), the speaker clustering part (yellow), and the set-specific DA classification part (red).

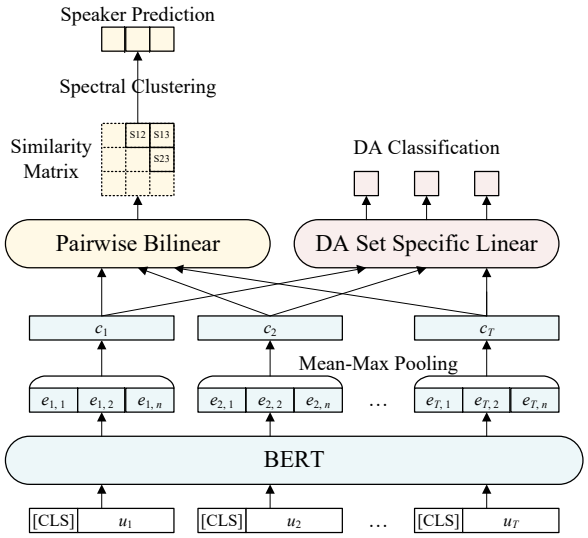

Figure 1: Model structure.

During training process, each data batch consists of B dialogues. To simplify the expression, we will omit the term of batch averaging in the following formulas. In a dialogue of the batch, there are T turns. The speaker of the i-th turn is s_i. The utterance of the i-th turn is u_i.

The objective of the model follows multi-task learning framework. The loss function of each data batch L is a combination of the binary cross entropy loss of the pairwise matrix L_{mat} and the cross entropy loss of the DA prediction L_{DA}. We use a hyperparameter λ to moderate the association between the two objectives. Formally,

$$L = L_{\text{mat}} + \lambda L_{\text{DA}}. \tag{1}$$

The objective and structure will be described in detail in the following sections.

3.1 Utterance Embedding

The first step of this model is to represent the semantic content of utterances as distributed vectors. Following previous works on text representation and dialogue processing (Ma et al., 2019; Gu et al., 2021), we concatenate the utterances in the dialogue with a [CLS] token prepended at the beginning of every utterance, and append a [SEP] token after them. For a dialogue with T utterances $u_1, u_2, \ldots, u_T$, the input format is

$$\text{[CLS]}\ u_1\ \text{[CLS]}\ u_2\ \ldots\ \text{[CLS]}\ u_T\ \text{[SEP]}.$$

Comparing to modeling each utterance in a separate pre-trained language model, this format is more lightweight that uses only a single BERT model, and contributes to directly calculate the word-level attentions across the utterances.

For each utterance, we take the output vectors of all the tokens (including the leading [CLS] token), and concatenate the mean pooling and max pooling results as the semantic representation, i.e., contextualized utterance embedding. Formally, the j-th token of the utterance u_i corresponds to the contextualized token embedding $e_{i,j}$ outputted by BERT. The utterance embedding is

$$c_i = \text{concat}\left[\underset{j}{\text{mean}}(e_{i,j}), \underset{j}{\max}(e_{i,j})\right], \tag{2}$$

where mean and max are mean pooling and max pooling functions through the stream dimension. For a BERT model of hidden size d_{BERT}, the length of the contextualized utterance embedding is $2d_{\text{BERT}}$.

3.2 Speaker Clustering

The relations between speakers are calculated by the form of bilinear. Specifically, for a dialogue with T turns, the contextualized utterance embeddings are

$$c_1, c_2, \ldots, c_T \in \mathbb{R}^{2d_{\text{BERT}}}.$$

The similarity score of the utterances u_m and u_n is the sigmoid mapping of bilinear form

$$\text{sim}(m, n) = \sigma(c_m^{\mathsf{T}} W c_n + b), \tag{3}$$

where $W \in \mathbb{R}^{2d_{\text{BERT}} \times 2d_{\text{BERT}}}$ and $b \in \mathbb{R}$ are trainable parameters.

For each pair of utterances, the similarity score is a real number between 0 and 1, denotes the probability that the corresponding speakers are identical.

The similarities are symmetric, so each pair of utterances is calculated just once, i.e., always having $m < n$ in Equation 3. All pairs of utterances finally form a symmetric $T \times T$ matrix.

The loss function of the matrix is calculated with the elements of the triangular. Formally,

$$L_{\text{mat}} = \frac{1}{C} \sum_{m=1}^{T-1} \sum_{n=m+1}^{T} \text{BCE}[\text{sim}(m, n), \mathbb{I}(s_m = s_n)], \quad (4)$$

where $C = T(T-1)/2$ is the number of the utterance pairs in the dialogue, $\mathbb{I}$ is indicator function, and BCE is Binary Cross Entropy loss function[1].

It is worth noticing that no additional positional encoding or embedding is added when calculating the similarity scores. We find that the positional information taken from BERT is enough for current calculation. Adding another positional information to this layer does not improve the performance according to our preliminary experiments.

We follow the spectral clustering algorithm to cluster the utterances into clusters that each cluster has the same speaker and different clusters have different speakers (Von Luxburg, 2007; Lin et al., 2019). For the audiences who are not familiar with this technique, spectral clustering is a clustering approach based on similarity graph and graph min-cut problem, which has nothing to do with speech spectra.

Given the symmetric similarity matrix $S \in \mathbb{R}^{T \times T}$, we compute both of the two kinds of normalized graph Laplacians, L_{sym} and L_{rw}, which are the same as the definition in the review (Von Luxburg, 2007). We use the eigenvalues of L_{rw} to determine the number of clusters, and the eigenvectors of L_{sym} to cluster[2].

The eigenvalues of the Laplacian matrix are related to the number of clusters. If the appropriate number of clusters is k, there will be a larger difference between the k-th smallest eigenvalue and the $(k+1)$-th smallest eigenvalue, which is known as the spectral gap. The greater the number of clusters, the less the overall eigenvalues will be. Therefore, an appropriate threshold can be selected on the validation set. If the k-th eigenvalue is greater than the threshold, the number of clusters will be considered to be less than k. Conversely, if the k-th eigenvalue is less than the threshold, the number of clusters will be considered to be greater than or equal to k. The threshold is adjusted on the validation set to maximize the accuracy. We report the results of both using the actual number of speakers as the number of clusters and using the spectral gap method to determine the number of clusters in the experiment section.

3.3 Auxiliary Set-specific Dialogue Act (DA) Classification

This part is designed as a auxiliary task to infuse dialogue act information into utterance embeddings. We assume that the ability of understanding semantic content will be stronger and the calculation of similarity will be more accurate if the model can predict the dialogue act of utterance correctly.

We present DA classification as part of the multi-task learning framework. For each dataset, if there are dialogue act annotations, we can use these labels to supervise the model to adjust the embeddings so that they express the corresponding dialogue acts. However, there is a problem that most of the DA-annotated datasets are not big enough, comparing to the speaker-annotated datasets. Meanwhile, these datasets are annotated with different sets and rules, and they are difficult to map to each other.

To solve this problem, we use a set-specific linear layer to adapt to different DA annotation sets. For different DA annotation sets, we use different linear layers to predict the corresponding number of dialogue act types. The loss function L_{DA} is calculated by the multi-class cross entropy[3] of the corresponding linear layer, and the output from other linear layers is ignored. With a shallow layer, we can expect to obtain a more general semantic representation. Formally,

$$L_{\text{DA}} = -\frac{1}{T} \sum_{m=1}^{T} \log \frac{\exp(z_{m,t_m})}{\sum_{d=1}^{D} \exp(z_{m,d})}, \quad (5)$$

where $z_{m,d}$ is the output of the set-specific linear layer of the m-th turn, d-th DA class, and t_m is the actual DA class of the m-th turn.

[1] For the definition, refer to: https://pytorch.org/docs/stable/generated/torch.nn.BCELoss.html .

[2] Implemented by scikit-learn and called with parameter `assign_labels="discretize"`.

[3] For the definition, refer to: https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html .

4 Experiment

4.1 Datasets

Our datasets are composed of three corpora: the Switchboard Dialogue Act Corpus (SwDA) (Stolcke et al., 2000), the Meeting Recorder Dialogue Act Corpus (MRDA) (Shriberg et al., 2004), and the Ubuntu Dialogue Corpus (Lowe et al., 2015). The SwDA Corpus and the MRDA Corpus are two common DA-annotated datasets. The SwDA Corpus is a two-party dialogue dataset transcribed by phone calls. The DA annotations are divided into 217 small categories and 43 major categories. The MRDA Corpus is a multi-party dialogue dataset transcribed by conferences. The DA annotations are divided into 52 full categories, 12 general categories, and 5 basic categories. The Ubuntu Dialogue Corpus is a widely used dialogue dataset collected from the chat records on the Ubuntu IRC system, without DA annotation. For all the three datasets, the adjacent utterances may be from the same speaker.

For the SwDA Corpus, we first split the dialogue streams into 10-turn segments, and then randomly divide them into training, validation and test set by the ratio of 8:1:1. For the MRDA Corpus, we use the same set division as the original data, and then split the dialogue streams into 10-turn segments. For the Ubuntu Dialogue Corpus, We use the 10-turn version released by previous works (Ouchi and Tsuboi, 2016; Gu et al., 2021). Table 1 shows the basic quantity statistics of the datasets.

Dataset	Set	Dialogues	S/D
SwDA	Train	17059	2.00
	Valid	2132	2.00
	Test	2132	2.00
MRDA	Train	7485	3.01
	Valid	1636	2.91
	Test	1664	2.96
Ubuntu	Train	495226	4.08
	Valid	30974	4.21
	Test	35638	4.19

Table 1: Statistics of the datasets. "S/D" stands for "average number of different Speakers per Dialogue".

In the experiments, we use the 43 major categories of SwDA and the 52 full categories of MRDA as our target DA sets in the auxiliary task.

We propose the results of the SwDA dataset and the MRDA dataset as DA-annotated single-corpus scenarios to analyze the role of the pairwise calculation and the auxiliary DA classification task, and the result of simultaneously training on SwDA, MRDA, and Ubuntu datasets as a sophisticated cross-corpus scenario. We will focus more on the experimental results on the MRDA dataset, because this dataset is both DA annotated and multi-party, which is convenient to analyze various aspects of the model.

4.2 Metrics

We employ two metrics in the experimental results, the adjusted Rand index (ARI) (Hubert and Arabie, 1985)[4] and the accuracy (ACC). The adjusted Rand index is a common metric for clustering, which measures the similarity between two sets of clusters. The value ranges from -1 to 1. For a random clustering, the mathematical expectation of ARI is 0. The accuracy is calculated by transforming the clustering problem into a classification problem. The idea is finding the best injective mapping from the predicted clusters to the actual clusters. Formally, enumerate all permutations of the set $\{1, 2, \ldots, n\}$ where n is the number of predicted clusters, so that

$$\mathrm{ACC}\left(y, \hat{y}\right) = \max_{p \in P} \frac{1}{T} \sum_{i=1}^{T} \mathbb{I}\left[p\left(\hat{y}^{(i)}\right) = y^{(i)}\right], \tag{6}$$

where y is the labels of actual clusters, $\hat{y}$ is the labels of predicted clusters, p is a permutation of the set $\{1, 2, \ldots, n\}$, $\mathbb{I}$ is indicator function, and $y^{(i)}$ is the element on index i in vector y.

The ACC result is utterance-level average statistics, which is the number of correctly cluster-assigned utterances divided by the total number of turns in the dataset. The ARI result is dialogue-level average statistics, which is the mean ARI values among the dialogues.

The reason for using accuracy as a metric is that it is convenient to observe the difference between the predicted speakers and the real speakers after mapping. And it provides a comparable result with other speaker identification models, not just speaker clustering models.

4.3 Setup

We use the PyTorch framework (Paszke et al., 2019) and common backpropagation for training. During

[4]Implemented by scikit-learn.

training, we calculate the metrics on the validation set and save the model parameters that maximize the accuracy on the validation set to avoid overfitting.

We use AdamW (Loshchilov and Hutter, 2019) as the optimizer. By validating on the SwDA dataset, we select the hyperparameters in lr={1×10^{-5}, 2×10^{-5}, 3×10^{-5}}, eps={1×10^{-4}, 1×10^{-5}, 1×10^{-6}}, and weight_decay={0, 1×10^{-4}}, to maximize the accuracy on the validation set. The final choice, lr=2×10^{-5}, eps=1×10^{-6}, weight_decay=0, betas=(0.9, 0.999), are used for all datasets.

We use BERT-base-uncased provided by Google (Devlin et al., 2019; Turc et al., 2019) as the initialization parameter of the BERT part. All of the BERT parameters and other linear and bilinear parameters are fine-tuned end-to-end.

For the SwDA and MRDA single-corpus experiments, we select the association hyperparameter in $\lambda = \{0.01, 0.05, 0.1, 0.2, 0.3, 0.5\}$. Every setting is trained on a single RTX 2080Ti GPU for about 1.5 hours to select the best one on the validation set. The final choice is $\lambda = 0.2$ for SwDA and $\lambda = 0.1$ for MRDA.

For the SwDA, MRDA, and Ubuntu cross-corpus experiment, we select the association hyperparameter in $\lambda = \{0.005, 0.01, 0.1\}$. For each training step, the data batch consists of 3 random Ubuntu dialogue segments, 1 random SwDA dialogue segment, and 1 random MRDA dialogue segment. Every setting is trained on a single RTX 2080Ti GPU for about 2 days to select the best one on the validation set. The final choice is $\lambda = 0.01$.

4.4 Baselines

Due to the lack of related works of text-based speaker clustering, we cannot find an existing model that is directly comparable. So we implement our baselines to prove the necessity of the model design.

The first design to test is modeling the pairwise relations. For comparison, we implemented a general sequence classification model that changes the speaker clustering part (including pairwise bilinear layer and similarity matrix layer) to a multi-class softmax layer. The number of output classes is set to the maximum number of different speakers in the dialogue. We trained this baseline model to predict the sequential IDs of speakers in a dialogue.

The second design to test is the set-specific dialogue act classification task. For comparison, we set $\lambda = 0$ as the ablation setting in this scenario, while other parameters including the constitution of input batches are consistent.

4.5 Results

Our experimental results of the single-corpus scenarios are shown in Table 2 and Table 3. The result of the cross-corpus scenario is shown in Table 4.

Model	Valid		Test	
	ACC	ARI	ACC	ARI
Baseline	.760	.486	.748	.463
Clustering	**.868**	**.596**	**.860**	**.575**
- w/o DA Task	.865	.585	.856	.566

Table 2: Result of SwDA dataset. The number of clusters is set to 2 as the consistent groud-truth.

Model	Valid		Test	
	ACC	ARI	ACC	ARI
Baseline	.543	.204	.527	.179
Clustering*	**.714**	**.317**	**.703**	**.301**
- w/o DA Task*	.706	.306	.700	.296
Clustering†	**.654**	**.298**	**.644**	**.279**
- w/o DA Task†	.648	.286	.642	.277

Table 3: Result of MRDA dataset. *: Given the actual number of different speakers in the dialogue as the number of clusters for spectral clustering. †: Using spectral gap method to predict the number of clusters for spectral clustering.

Table 2 and Table 3 show the results of SwDA and MRDA datasets respectively. Our multi-task clustering model outperforms the sequence classification baseline and the ablative setting without auxiliary DA classification task in all the tests. These results prove that our auxiliary task improves the semantic content representation and similarity calculation if the training data has DA annotation and the evaluating data has the same distribution as the training data. The result of using the spectral gap method to detect the number of clusters shows that this model still outperforms the baseline and the ablative setting even without prior knowledge of the actual number of clusters.

Table 4 shows the results of training on all of the three datasets, and evaluating on either all three datasets or just the Ubuntu datasets. This model still outperforms the baseline in all the tests. It also outperforms the ablative setting in all the tests in the scenario of given the ground-truth number

Model Structure	S+M+U				S+M+U (Ubuntu Only)			
	Valid		Test		Valid		Test	
	ACC	ARI	ACC	ARI	ACC	ARI	ACC	ARI
Baseline	.530	.249	.531	.247	.513	.234	.516	.235
Clustering*	**.697**	**.299**	**.695**	**.296**	**.685**	**.279**	**.685**	**.280**
- w/o DA Task*	.696	.297	.694	.292	.684	.277	.684	.277
Clustering†	.632	**.284**	.631	**.282**	**.618**	**.264**	**.619**	.264
- w/o DA Task†	**.633**	.283	**.632**	.281	**.618**	**.264**	**.619**	**.265**

Table 4: Result of training synergistically on SwDA, MRDA, and Ubuntu datasets, and evaluating on the three datasets (left) or only the Ubuntu dataset (right). *: Given the actual number of different speakers in the dialogue as the number of clusters for spectral clustering. †: Using spectral gap method to predict the number of clusters for spectral clustering.

of speakers. Even the Ubuntu-only result is promoted by our set-specific DA classification task. This proves that cross-corpus supervised training is possible if we design the model with reasonable structure and objective.

Another phenomenon reflected in Table 4 is that, without specifying real number of speakers, there is a different trend between the results of ACC metric and ARI metric on S+M+U data. Actually, ARI is more concerned about whether the dividing points of clusters are correct, while ACC is the result after mapping. Therefore, ARI is a more direct metric that indicates whether the key points of speaker alternation are found correctly.

5 Discussion

In this section, we discuss whether the substructures of the model work accurately, and whether the model is disturbed by some possible factors (semantic similarity and utterance distance).

5.1 Determining the Number of Speakers

For clustering problems, it is an important step to predict an appropriate number of clusters. The principle of using spectral gap to predict the number of clusters has been described in Section 3.2. In order to verify whether this method can accurately predict the number of clusters, we make statistics on the MRDA dataset. We also tried training a multilayer perceptron (MLP) with the eigenvalues to predict the number of clusters. The multilayer perceptron uses 90% of the validation set data for training and the remaining 10% for validation.

Table 5 shows that the spectral gap method can predict more than 94% of the test data almost accurately, where error is less than or equal to 1. This method is more accurate than the multilayer perceptron.

Method	Accurate		$\leq \pm 1$	
	Valid	Test	Valid	Test
Spectral Gap	.485	**.482**	.941	**.942**
- w/o DA Task	.464	.468	.941	.941
MLP	.498	.480	.927	.928
- w/o DA Task	.499	.467	.928	.931

Table 5: Speaker (cluster) number prediction accuracy on the MRDA dataset. "$\leq \pm 1$" means the proportion of data whose difference between the predicted value and the actual value is less than or equal to 1.

The result also shows that the DA auxiliary task can not only directly improve the accuracy of speaker relation detection, but also help improve the accuracy of speaker number prediction.

5.2 Distinguishing Speaker Relation and Semantic Similarity

The similarity calculation takes a bilinear form. In this case, it is necessary to check whether the model confuses speaker similarity and utterance text similarity (semantic similarity). Semantic similarity is one of the fundamental features for inferring the relation between speakers, i.e., utterances from the same speaker tend to be semantically similar (Kundu et al., 2012; Ma et al., 2017; Ek et al., 2018). However, it would be harmful if the model takes semantic similarity as the only factor in prediction, because utterances from different speakers with same words are very common in dialogues, such as greetings, farewells, and rhetorical questions, and they will make a higher rate of false positives. Therefore, it is necessary to prove that the utterance embeddings and the similarity scores take full account of the contextual utterances, instead of simply extracting context-independent semantic features of the utterances.

	Accurary	Student's t-test p-value	Spearman's rank
Correctness	P1−S1 = **0.645**	P2−S1 = $\mathbf{4.49 \times 10^{-104}}$	-
Confusion	P1−C1 = 0.550	P1−C2 = 8.66×10^{-12}, P2−C1 = 9.70×10^{-4}	P2−C2 = 0.0866
Inherence	S1−C1 = 0.526	S1−C2 = 1.23×10^{-3}	-

Table 6: The correlations about correctness (between P and S), confusion (between P and C), and inherence (between S and C). There is no Spearman's rank correlation coefficient about correctness or inherence because S does not have numerical dimension.

The way of demonstration is calculating three types of correlations:

- **Correctness**: The correlation between prediction results (P) and speaker relations (S).
- **Confusion**: The correlation between prediction results (P) and context-independent semantic similarities (C).
- **Inherence**: The correlation between speaker relations (S) and context-independent semantic similarities (C).

If the correctness is much greater than the confusion, it will prove that the model is aware of speaker relations without being compromised by context-independent semantic similarity. The inherence is necessary because the speakers and the semantic features are dependent, and the ground correlation between them needs to be excluded.

To determine these three types of correlations, we collected values in 5 dimensions:

- P1: The prediction result of whether the speakers are same or different (binary values).
- P2: The prediction result of similarity score in the pairwise matrix (numerical values).
- S1: Whether two speakers are same of different (binary values).
- C1: Whether two utterances are semantically similar or dissimilar (binary values).
- C2: The semantic similarity between two utterances (numerical values).

The values are collected from the validation set of MRDA. We select one pair of turns with the same speaker and one pair of turns with different speakers from each dialogue to form a new dataset. In this dataset, the two types of S1 are balanced. The utterance embeddings are calculated with a pre-trained-only BERT model[5], and the values in C2 are calculated by cosine similarity between the embeddings. Then, the pairs of turns are sorted by C2 in ascending order, and the first half of the pairs are regarded as dissimilar pairs, and the last half of the pairs are regarded as similar pairs, forming a balanced C1. The values in P1 and P2 are predicted by the model.

The experimental results are divided into three categories:

- The correlation between two binary dimensions is evaluated by the accuracy (whether it meets the hypothesized association). A greater value indicates a stronger correlation.
- The correlation between a binary dimension and a numerical dimension is evaluated by Student's t-test. A smaller p-value indicates a stronger correlation.
- The correlation between two numerical dimensions is evaluated by Spearman's rank correlation coefficient. A greater absolute value indicates a stronger correlation.

The statistics of correlation are shown in Table 6. Three conclusions can be drawn from it:

First, there is a strong correlation between prediction results and speaker relations. As the Correctness row shows, P1−S1 is much greater than 0.5, and P2−S1 is very small.

Second, there is a ground correlation between speaker relations and context-independent semantic similarities. As the Inherence row shows, S1−C1 is slightly greater than 0.5, and S1−C2 is between 10^{-3} and 10^{-2}.

Third, there is a weak correlation between prediction results and context-independent semantic similarities. As the Confusion row shows, P1−C1 is slightly greater than 0.5, P1−C2 and P2−C1 are less than 10^{-3}, and P2−C2 is slightly greater than but close to 0. But this correlation is mainly

[5]BERT-base-uncased without any fine-tuning.

brought by the ground correlation between speaker relations and semantic similarities, because the correlations about Confusion approximately equal to the correlations about Inherence, and much less than the correlations about Correctness.

These results complete the demonstration that this model can detect speaker relations without being compromised by context-independent semantic similarity.

5.3 Distance Impacts Similarity Modeling

We investigate the results of internal layer of the pairwise similarity score by aggregating the position-level error of similarity matrix, as shown in Figure 2. The item in m-th row and n-th column is the mean error of the similarity score of the m-th turn and n-th turn. Formally,

$$\mathrm{err}(m,n) = \left|\mathrm{sim}(m,n) - y(m,n)\right|. \quad (7)$$

We take the results of similarity matrix on the Ubuntu test set, and plot the heatmap of mean error. The figure shows that the model successfully models the relations between utterances, especially the adjacent ones. For longer-distance pairs, it is constitutionally more difficult to be modeled, but the model is still effective with a mean error less than 0.5.

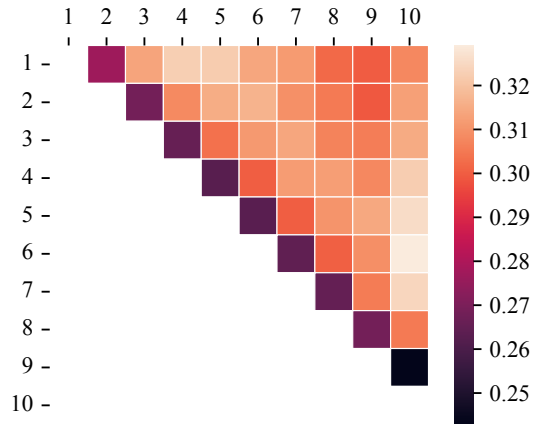

Figure 2: Error heatmap of similarity matrix on the Ubuntu test set with auxiliary DA classification task. Darker color means more accurate, and lighter color means more erring.

6 Conclusion

We propose a text-based dialogue speaker clustering model. Based on the theory of the dialogue structure, the model holds the semantic content and the communicative functions explicitly with the BERT layer and the similarity matrix. The model is enhanced by the idea of cross-corpus supervision with the auxiliary set-specific dialogue act classification task. It finally generates the cluster labels of speakers with spectral clustering. Our model outperforms the sequence classification baseline and the non-DA ablation on almost all tests. Additional discussion illustrates the accuracy in predicting the number of speakers (clusters) and the ability to distinguish between speaker relation and semantic similarity of our model. We also show that the precision of speaker similarity prediction varies with utterance distance.

In future research, it is worth trying further pretraining the model on dialogue data, which will likely help to perceive dialogue turns and extract better utterance embeddings. We will also explore for a method to make the similarity calculation between long-distance utterance pairs more accurate.

References

Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a human-like open-domain chatbot. *CoRR*, abs/2001.09977.

Zakaria Aldeneh, Dimitrios Dimitriadis, and Emily Mower Provost. 2018. Improving end-of-turn detection in spoken dialogues by detecting speaker intentions as a secondary task. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6159–6163.

Xavier Anguera, Simon Bozonnet, Nicholas Evans, Corinne Fredouille, Gerald Friedland, and Oriol Vinyals. 2012. Speaker diarization: A review of recent research. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(2):356–370.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Harry Bunt, Jan Alexandersson, Jean Carletta, Jae-Woong Choe, Alex Chengyu Fang, Koiti Hasida, Kiyong Lee, Volha Petukhova, Andrei Popescu-Belis,

Laurent Romary, Claudia Soria, and David Traum. 2010. Towards an ISO Standard for Dialogue Act Annotation. In *Seventh conference on International Language Resources and Evaluation (LREC'10)*, La Valette, Malta.

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

Adam Ek, Mats Wirén, Robert Östling, Kristina N. Björkenstam, Gintarė Grigonytė, and Sofia Gustafson Capková. 2018. Identifying speakers and addressees in dialogues extracted from literary fiction. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Yihui Fu, Luyao Cheng, Shubo Lv, Yukai Jv, Yuxiang Kong, Zhuo Chen, Yanxin Hu, Lei Xie, Jian Wu, Hui Bu, Xin Xu, Jun Du, and Jingdong Chen. 2021. AISHELL-4: An Open Source Dataset for Speech Enhancement, Separation, Recognition and Speaker Diarization in Conference Scenario. In *Proc. Interspeech 2021*, pages 3665–3669.

Jia-Chen Gu, Chongyang Tao, Zhenhua Ling, Can Xu, Xiubo Geng, and Daxin Jiang. 2021. MPC-BERT: A pre-trained language model for multi-party conversation understanding. 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 3682–3692, Online. Association for Computational Linguistics.

Sibylle Hess, Wouter Duivesteijn, Philipp Honysz, and Katharina Morik. 2019. The spectacl of nonconvex clustering: A spectral approach to density-based clustering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3788–3795.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of classification*, 2(1):193–218.

Naoyuki Kanda, Xiong Xiao, Yashesh Gaur, Xiaofei Wang, Zhong Meng, Zhuo Chen, and Takuya Yoshioka. 2022. Transcribe-to-diarize: Neural speaker diarization for unlimited number of speakers using end-to-end speaker-attributed asr. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8082–8086.

Amitava Kundu, Dipankar Das, and Sivaji Bandyopadhyay. 2012. Speaker identification from film dialogues. In *2012 4th International Conference on Intelligent Human Computer Interaction (IHCI)*, pages 1–4.

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

Ruizhe Li, Chenghua Lin, Matthew Collinson, Xiao Li, and Guanyi Chen. 2019. A dual-attention hierarchical recurrent neural network for dialogue act classification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 383–392, Hong Kong, China. Association for Computational Linguistics.

Wei Li and Yunfang Wu. 2016. Multi-level gated recurrent neural network for dialog act classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1970–1979, Osaka, Japan. The COLING 2016 Organizing Committee.

Yuhai Liang and Qiang Zhou. 2020. Detect turn-takings in subtitle streams with semantic recall transformer encoder. In *2020 International Conference on Asian Language Processing (IALP)*, pages 1–6.

Qingjian Lin, Ruiqing Yin, Ming Li, Hervé Bredin, and Claude Barras. 2019. LSTM Based Similarity Measurement with Spectral Clustering for Speaker Diarization. In *Proc. Interspeech 2019*, pages 366–370.

Pierre Lison, Jörg Tiedemann, and Milen Kouylekov. 2018. OpenSubtitles2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The Ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 285–294, Prague, Czech Republic. Association for Computational Linguistics.

Yanick Lukic, Carlo Vogt, Oliver Dürr, and Thilo Stadelmann. 2016. Speaker identification and clustering using convolutional neural networks. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6.

Kaixin Ma, Catherine Xiao, and Jinho D. Choi. 2017. Text-based speaker identification on multiparty dialogues using multi-document convolutional neural networks. In *Proceedings of ACL 2017, Student Research Workshop*, pages 49–55, Vancouver, Canada. Association for Computational Linguistics.

Xiaofei Ma, Zhiguo Wang, Patrick Ng, Ramesh Nallapati, and Bing Xiang. 2019. Universal text representation from BERT: an empirical study. *CoRR*, abs/1910.07973.

Kai Nickel, Tobias Gehrig, Rainer Stiefelhagen, and John McDonough. 2005. A joint particle filter for audio-visual speaker tracking. In *Proceedings of the 7th International Conference on Multimodal Interfaces*, ICMI '05, page 61–68, New York, NY, USA. Association for Computing Machinery.

Hiroki Ouchi and Yuta Tsuboi. 2016. Addressee and response selection for multi-party conversation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2133–2143, Austin, Texas. Association for Computational Linguistics.

Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J. Han, Shinji Watanabe, and Shrikanth Narayanan. 2022. A review of speaker diarization: Recent advances with deep learning. *Computer Speech & Language*, 72:101317.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Desh Raj, Pavel Denisov, Zhuo Chen, Hakan Erdogan, Zili Huang, Maokui He, Shinji Watanabe, Jun Du, Takuya Yoshioka, Yi Luo, Naoyuki Kanda, Jinyu Li, Scott Wisdom, and John R. Hershey. 2021. Integration of speech separation, diarization, and recognition for multi-speaker meetings: System description, comparison, and analysis. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 897–904.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325, Online. Association for Computational Linguistics.

Emanuel A. Schegloff and Harvey Sacks. 1973. Opening up closings. *Semiotica*, 8(4).

John R Searle and John Rogers Searle. 1969. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press.

Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang, and Hannah Carvey. 2004. The ICSI meeting recorder dialog act (MRDA) corpus. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*, pages 97–100, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–374.

Dinoj Surendran and Gina-Anne Levow. 2006. Dialog act tagging with support vector machines and hidden markov models. In *Ninth International Conference on Spoken Language Processing*.

S.E. Tranter and D.A. Reynolds. 2006. An overview of automatic speaker diarization systems. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5):1557–1565.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: The impact of student initialization on knowledge distillation. *CoRR*, abs/1908.08962.

Félicien Vallet, Jim Uro, Jérémy Andriamakaoly, Hakim Nabi, Mathieu Derval, and Jean Carrive. 2016. Speech trax: A bottom to the top approach for speaker tracking and indexing in an archiving context. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2011–2016, Portorož, Slovenia. European Language Resources Association (ELRA).

Ulrike Von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

Ce Wang and M.S. Brandstein. 1999. Multi-source face tracking with audio and visual data. In *1999 IEEE Third Workshop on Multimedia Signal Processing (Cat. No.99TH8451)*, pages 169–174.

Han Xiao. 2018. bert-as-service. `https://github.com/hanxiao/bert-as-service`.

Xuejing Zhang, Xueqiang Lv, and Qiang Zhou. 2018. Chinese dialogue analysis using multi-task learning framework. In *2018 International Conference on Asian Language Processing (IALP)*, pages 102–107.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.