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

Anonymous ACL submission

Abstract

We propose a textual dialogue speaker clustering model, which groups the utterances of a multi-party dialogue without speaker annotations, so that the real speakers are identical inside each cluster. We find that, even without knowing the speakers, the interactions between utterances are still implied in the text. Such interactions suggest the correlations of the speakers. In this work, we model the semantic content of an utterance with a pre-trained language model, and the correlations of speakers with an utterance-level pairwise matrix. The semantic content representation can be further enhanced by additional cross-corpus supervised dialogue act modeling. The speaker labels are finally generated by spectral clustering. Experiment shows that our model outperforms the sequence classification baseline, and benefits from the set-specific dialogue act classification auxiliary task. We also discuss the detail of correlation modeling and step-wise training process.

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-oriented models are designed to input speaker information, i.e., speaker embeddings or just assuming the dialogue is composed of two speakers involved turn by turn, to introduce dialogue structure information to the models. However, for common researchers, dialogue data is hard to collect. Most of big dialogue data used by enterprises cannot be published due to privacy and legal issues. Although there are some open dialogue data, they are either not sufficient enough, or task-oriented, or lack of speakers labeling, such as OpenSubtitles (Lison et al., 2018). Datasets like subtitles contain a lot of dialogue data of daily communication, but due to the lack of speaker annotation, it is difficult to be utilized in current dialogue models. Some researches in related fields, such as multimodal speech processing, may also need text-based speaker clustering techniques. So it is valuable to develop a model to reconstruct the missing identities of speakers in such dialogue data.

In order to reconstruct the speaker labels in the dialogue, our work is dedicated to the method of speaker clustering. Different from previous researches on speaker identification, which aim at selecting the most similar speaker of each utterance from the known 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 closely related to the dialogue structure, because the process of turns follows certain patterns. These patterns include the change of speakers, also known as turn-takings, and the interactions between different speakers. Conversely, speaker relations will be available if we get these patterns from textual utterances. It is intuitively possible, because the turn-takings can be detected by some methods like the language model, and the interactions can be detected by word-level features and utterance-level acts such as greetings, questions, and responses. Other semantic features like interruptions and coreferences are also helpful.

To achieve the above dialogue analysis process, we need to analyze the roles of utterances in a dialogue. The first role is that an utterance carries semantic content with its text form. Many conventional dialogue comprehension studies take dialogue act (DA) classification as a target task. We suggest that this conventional task helps the pre-trained model to embed the semantic content of utterances, because the most important thing in

embedding learning is to make the representation have a good distribution which expresses the corresponding features of the embedded item. The second role of the turn is that it has communicative functions, which are mainly under covered by the correlations between turns. We can explicitly model these communicative functions by pairwise calculation.

Every communicative function is related to a correlation between speakers. For both bi-party and multi-party dialogues, the correlation has only two possible values, i.e., whether same or different. The correlations among the whole dialogue can form a correlation matrix, which is regarded as the similarity graph of the ground speakers. The graph can reconstruct the clusters of speakers with a density-based clustering method. The most popular algorithms incorporating these paradigms 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 our task that each utterance must be in a cluster.

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

Our model is fine-tuned and evaluated on the Switchboard Dialog 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 experimental result proves the significance of our pairwise correlation design and cross-corpus dialogue act classification auxiliary task.

2 Related Works

Our task is related to dialogue comprehension. Generally, we start with the dialogue structure, and look for methods of semantic content modeling, communicative function modeling, and speaker clustering.

Dialogue structure: The traditional researches in dialogue processing have noticed that the 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 are represented by dialogue acts (Searle and Searle, 1969) and adjacency pairs (Schegloff and Sacks, 1973) respectively. Every turn has its own dialogue act. Two turns of different speakers form a 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., 2018a). Speaker clustering is strongly depended on dialogue structure because the semantic content and communicative functions involve the correlations of speakers.

Semantic content modeling: 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).

Recently there are some dialogue comprehension models based on pre-trained language models. Most of them are trained to extract semantic representations in a self-supervised manner (Zhang and Zhao, 2021). The advantage is that the available data is rich and the task is adaptable. However, there are still some data that are not large in amount, but clearly and instructively labeled, such as dialogue act annotated corpus. We want to use these annotated data, although their annotations may not be the same annotation set, to form a cross-corpus supervised training.

Communicative function modeling: Previous works on communicative function are almost detecting the adjacency pairs of close turns (Nakanishi et al., 2019; Zhang et al., 2018b). But these works cannot be directly used in our task, because they depend on and aim at the annotations of adjacency pairs instead of speakers. We need a more general type of relations that coordinates with the correlation of speakers. Thus, we look for a general form of the relationship between two embeddings.

Speaker clustering: As far as we know, there are few works directly on speaker clustering in

textual dialogues. However, there is a previous work on speaker clustering through pairwise relationships based on speech signals (Lin et al., 2019). This work uses spectral clustering as the top-level structure. It provides an idea for our structural design, but its basic input data is voice rather than text.

Other models related to speakers in dialogue: There are some researches more related to the speaker labeling task in textual dialogues (Ek et al., 2018; Serban and Pineau, 2015; Ma et al., 2017), but they are not speaker clustering models directly. Most of them depend on the assumption that each speaker has its own speaking characteristic, 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. This kind of works has strengthened the feature extraction between turns and raised the performance to a high level. Although the result of turn-taking detection can be used to do speaker clustering in dialogues with only two speakers, it cannot be directly used in dialogues with more than three speakers (multi-party dialogues). The reason is that it only focuses on the relationships between two adjacent utterances. It would be useful in multi-party dialogues if extended to the relationships between utterances that are not necessarily adjacent.

3 Model

The main consideration of our model is how to get representation of utterances, cooperate with cross-corpus DA supervision, and calculate the correlations. Therefore, the model is divided into three parts in general: the BERT embedding part, the set-specific dialogue act classification part, and the correlation clustering part. Figure 1 shows the overall structure of our model.

We define that, during training process, each data batch consists of B dialogues. In the i-th dialogue of the batch, there are T_i turns. The speaker of the j-th turn of the i-th dialogue is $s_{i,j}$. The utterance text of the j-th turn of the i-th dialogue is $u_{i,j}$. The DA set of the i-th dialogue is D_i, where $D_i = \emptyset$ if the i-th dialogue does not have DA annotation. The DA label of the j-th turn of the i-th dialogue is $d_{i,j}$, where $d_{i,j} \in D_i$.

The objective of the model tallies with 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,

$$L = L_{\text{mat}} + \lambda L_{\text{DA}}. \quad (1)$$

The objectives and structure will be described in the following sections.

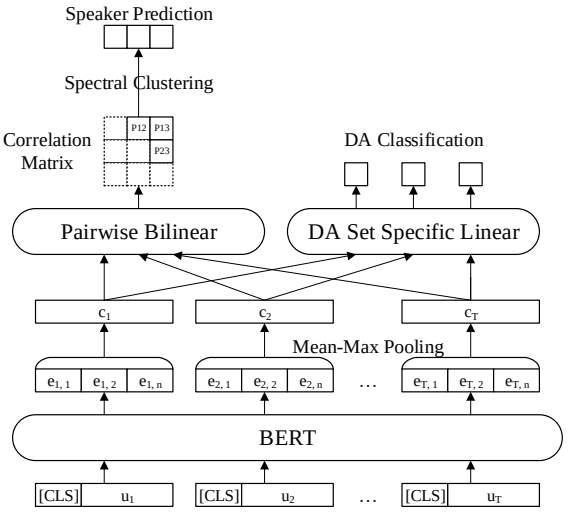

Figure 1: Model structure.

3.1 Utterance Embedding

The first step of our 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. For a dialogue with T_i utterances $u_{i,1}, u_{i,2}, \ldots, u_{i,T_i}$, the input format is

$$\text{[CLS]}\ u_{i,1}\ \text{[CLS]}\ u_{i,2}\ \ldots\ \text{[CLS]}\ u_{i,T_i}\ \text{[SEP]}.$$

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 modeling the word-level relations across the utterances. But it causes a problem if some utterances are too long. We deal with this problem by setting a limitation of the count of tokens for each utterance. The utterances which are longer than the limitation will be cut off. This scenario rarely happens so it will not impact our analysis.

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

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

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

3.2 Correlation Calculation

To model the correlations between speakers, we use the form of bilinear. Specifically, for a dialogue with T_i turns, the contextualized utterance embeddings are

$$c_{i,1}, c_{i,2}, \ldots, c_{i,T_i} \in \mathbb{R}^{2d_{\text{BERT}}}.$$

The correlation score of the utterances $u_{i,m}$ and $u_{i,n}$ is the sigmoid mapping of bilinear form

$$\text{corr}(m, n \mid i) = \sigma(c_{i,m}^{\mathsf{T}} W c_{i,n} + b), \quad (3)$$

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 correlation is a real number between 0 and 1, denotes the probability that the corresponding speakers are identical. The correlations are symmetric, so each pair of utterances is just calculated once, i.e., always having $m < n$ in Equation 3. All correlations in the i-th dialogue finally form a symmetric $T_i \times T_i$ matrix.

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

$$L_{\text{mat}} = \frac{1}{C} \sum_{i=1}^{B} \sum_{m=1}^{T_i - 1} \sum_{n=m+1}^{T_i} \text{BCE}[\text{corr}(m, n \mid i), y(m, n \mid i)], \quad (4)$$

$$y(m, n \mid i) = \mathbb{I}[s_{i,m} = s_{i,n}], \quad (5)$$

where $C = \sum_{i=1}^{B} T_i(T_i - 1)/2$ is the number of the utterance pairs in the batch, and $\mathbb{I}$ is indicator function.

The reason why we can directly use boolean values as the correlation classes is that the correlation classes are supersets of dialogue structure types. For example, the non-identical correlation (0) is the union of the scenarios of turn-taking, forming an adjacency pair, and a chain of odd number of adjacency pairs, etc. While the identical correlation (1) is the union of the scenarios of non-turn-taking, forming a long-distance relation of restatement, continuous probing, and forming a chain of even number of adjacency pairs, etc. It is possible to detect the turn-taking states and the dialogue structure relations as many previous works have shown, so classifying on the supersets is also possible.

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

3.3 Speaker Clustering

We follow the spectral clustering algorithm (Von Luxburg, 2007; Lin et al., 2019) to cluster the utterances into clusters that each cluster has the same speaker and different clusters have different speakers.

Given the symmetric correlations matrix $S \in \mathbb{R}^{T_i \times T_i}$, 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[1].

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, called the eigengap. 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 $(k+1)$-th eigenvalue is greater than the threshold, the number of clusters is considered to be (or less than) k. The threshold is adjusted on the validation set to maximize the accuracy. We report the results of both using the real number of

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

speakers as the cluster number and using the threshold method to determine the cluster number in the experiment section.

3.4 Set-specific Dialogue Act Classification

This part is designed as a auxiliary task to infuse dialogue act information into utterance embeddings. We suggest that the ability of extracting semantic content will be more strong and the calculation of correlation will be more accurate if the model can judge the dialogue act of the utterance correctly.

We present dialogue act classification as part of a multi-task learning framework. For each dataset, if there are dialogue act labels annotated, 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. Although the number of DA-annotated datasets is large, these datasets are annotated with different sets and rules, and cannot be easily mapped to each other.

To solve this problem, we use a set-specific linear layer to adapt to the DA set of the data. For different dialogue act 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 of the output of the corresponding linear layer, and the output of other linear layers is ignored.

This layer uses a shallow structure to classify in order to integrate different DA labeling rules and sets together. Ideally, this layer should be able to become a labeling rule converter, and make the contextualized utterance embedding connotes a more general dialogue act type.

4 Experiment

4.1 Datasets

Our datasets are composed of three corpus: the SwDA Corpus, the MRDA Corpus, and the Ubuntu Dialogue Corpus. We propose the result of the SwDA dataset as a simple single-corpus condition to intuitively analyze the role of correlation matrix and auxiliary DA classification task, and the result of simultaneously training on SwDA, MRDA, and Ubuntu datasets as a mass cross-corpus condition.

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. There are 221616 turns in total. The dialogue act annotations are divided into 217 small categories and 43 major categories. Two adjacent utterances may be from the same speaker. The MRDA Corpus is a multi-party dialogue dataset transcribed by conferences. There are 108202 turns in total. DA annotations are divided into 52 full categories, 12 general categories, and 5 basic categories. Two adjacent utterances may be from the same speaker, and this situation is relatively common. The Ubuntu Dialogue Corpus is a widely used dialogue dataset collected from the chat records on the Ubuntu IRC system, without DA annotation.

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). Each dialogue contains 10 turns, with the number of speakers ranging from 2 to 10. Table 1 shows the basic quantity statistics of the datasets.

Dataset	Set	Dialog	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 our experiment, we use the 43 major categories of SwDA and 12 general categories of MRDA as our target DA sets of the auxiliary task.

4.2 Parameters and Environment

We use the PyTorch framework (Paszke et al., 2019) and common backpropagation for training. After each epoch of training, we calculate the performance indicators 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 testing on the SwDA dataset, we select the hyperparameters in lr={1e-5, 2e-5, 3e-5}, eps={1e-4, 1e-5, 1e-6}, and weight_decay={0, 1e-4}, to maximize the accuracy on the validation set. The final choice, lr=2e-5, eps=1e-6, weight_decay=0, betas=(0.9, 0.999), are used for all datasets.

We use bert-base-uncased provided by Google (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 single-corpus experiment, we select the association hyperparameter in $\lambda = \{0.01, 0.05, 0.1, 0.2, 0.3, 0.5\}$. For each training step, the data batch consists of 5 random SwDA dialogue segments. The evaluation is executed for every 500 steps. 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 the SwDA, MRDA, and Ubuntu cross-corpus experiment, we select the association hyperparameter in $\lambda = \{0.01, 0.1\}$. 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. The evaluation is executed for every 5000 steps. 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 $\lambda = 0.01$.

4.3 Baselines

Due to the lack of related works on 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 cover is modeling the pairwise correlations. For comparison, we implemented a general sequence classification model that changes the output 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 cover 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.

4.4 Metrics

We employ two metrics in the experimental results, the adjusted Rand index (ARI) (Hubert and Arabie, 1985)[2] and the accuracy (ACC). The adjusted Rand index is a common metric for clustering, which measures the similarity between two cluster sets. The value ranges from -1 to 1. For a random clustering, the mathematical expectation of ARI is 0, which is intuitively correct. 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 real clusters. If the number of the predicted clusters is greater than the number of the real clusters, the exceeded predicted clusters will be mapped to nothing. Formally, we enumerate all permutations of the set $\{1, 2, \ldots, n\}$ where n is the number of predicted clusters, so that

$$\text{acc}\,(y, \hat{y}) = \max_{p \in P} \frac{1}{T_i} \sum_{j=1}^{T_i} \mathbb{I}\left[p\left(\hat{y}^{(j)}\right) = y^{(j)}\right],$$

where y is the labels of real 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 $y^{(j)}$ is the element on index j in vector y.

The ACC result is turn-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 result and the real value after mapping. And it provides a comparable result with other speaker identification models, not just speaker clustering models.

4.5 Result

Our experimental result of the single-corpus condition and the cross-corpus condition are shown in Table 2 and Table 3 respectively.

Table 2 shows the result of SwDA dataset. Our multi-task clustering model outperforms the sequence classification baseline and the ablative experiment without auxiliary DA classification task in all the tests. This result proves that our auxiliary

[2]Implemented by scikit-learn.

Model	SwDA			
	Valid		Test	
	ACC	ARI	ACC	ARI
Baseline	0.760	0.486	0.748	0.463
Clustering	**0.868**	**0.596**	**0.860**	**0.575**
- w/o DA Task	0.865	0.585	0.856	0.566

Table 2: Result of SwDA dataset.

task improves the semantic content representation and correlation calculation if the training data and evaluating data have the same semantic feature and distribution.

Table 3 shows the result of training on all of the three datasets, and evaluating on the three datasets or just on the Ubuntu datasets. Because MRDA and Ubuntu datasets are multi-party dialogues and we may not know the number of speakers in a dialogue in advance, we provide the result of both using the ground-truth speaker count as the number of clusters, and using the eigengap method to detect the number of clusters for spectral clustering. Our model still outperforms the baseline in all the tests, and the ablative experiment in almost all the tests in the scenario of given the ground-truth speaker count. Even the Ubuntu only result is promoted by our set-specific dialogue act classification task. This suggests that cross-corpus and cross-domain supervised training is possible if we design the model with reasonable structure and objective. The result of using the eigengap method to detect the number of clusters shows that our model still outperforms the baseline in all the tests, which suggests that the clustering method is still better than the sequence classification method even without prior knowledge of the real number of clusters.

An interesting phenomenon is that, without specifying the real number of speakers, the ACC metrics and ARI metrics of whether applying auxiliary tasks or not have different trends. Actually, the ARI metrics is more concerned with whether the dividing points of the clusters are correct, while the ACC metrics is the result after mapping. So the ARI metrics more directly reflects whether the key transformation relationships are correctly found.

5 Discussion

5.1 Correlation Modeling

The essential of speaker clustering task is modeling the dialogue structure, especially the interactions among the speakers. We have described the steps of speaker clustering are:

1. Observe the sequential utterance stream.
2. Extract the semantic content.
3. Infer the communicative functions (interactions and correlations).
4. Infer the identities of the speakers.

The third step is necessary and hard to be implicitly learnt by the sequential model. Pairwise correlations are suitable and sufficient to cover all the communicative functions in most cases, because a turn of the dialogue is an interaction between two speakers in most of time.

The sample output of the sequence classification model shows the necessity of modeling the pairwise correlations. We find that the results of the sequence classification models have a typical kind of error. It is that the model sometimes generates roughly segmented results, i.e., the first few utterances are predicted to be Speaker 0, the next few utterances are predicted to be Speaker 1, and the next few utterances are predicted to be Speaker 2, and so on. We have also tried some other sequence classification models and the results are similar. This shows that the model is in a state of underfitting, and the prediction results only satisfy the statistical characteristics along the stream dimension, but not the turn-taking characteristics along the semantic dimension.

One of the biggest difference between the speaker clustering task and other classification problems is that there is no direct statistical relationship between the input feature and the output classification. The speaker label of a turn is 0, 1, or 2, not because the turn itself has the feature of class 0, 1, or 2, but mainly because of its position in the dialogue. Therefore, the ordinary sequence classification model is more likely to learn the distribution of labels in the time dimension, but it is difficult to find the relationship between the same label or the relationship between different labels. The correlation based model avoids such problem.

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

$$\mathrm{err}(m,n) = \left|\mathrm{corr}(m,n) - y(m,n)\right|. \quad (6)$$

Model	S+M+U				S+M+U (Ubuntu Only)			
	Valid		Test		Valid		Test	
	ACC	ARI	ACC	ARI	ACC	ARI	ACC	ARI
Baseline	0.530	0.249	0.531	0.247	0.513	0.234	0.516	0.235
Clustering	**0.697**	**0.297**	**0.695**	**0.294**	**0.684**	0.276	**0.685**	**0.278**
- w/o DA Task	0.696	**0.297**	0.694	0.292	**0.684**	**0.277**	0.684	0.277
Clustering without Speaker Count	0.629	**0.285**	0.629	**0.283**	0.615	**0.265**	0.617	**0.267**
- w/o DA Task	**0.633**	0.283	**0.632**	0.281	**0.618**	0.264	**0.619**	0.265

Table 3: Result of training on SwDA, MRDA, and Ubuntu datasets, and evaluating on the three datasets (left) or only the Ubuntu dataset (right).

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

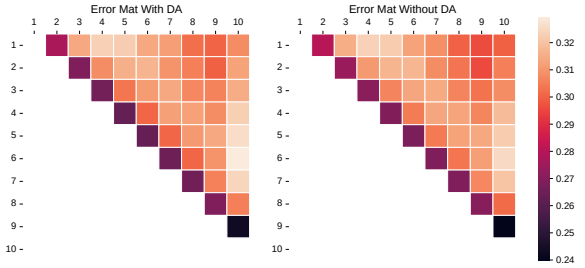

Figure 2: Error heatmaps of correlation matrix on the Ubuntu test set with (left) and without (right) auxiliary DA classification task. Darker color means more accurate, and lighter color means more erring.

5.2 Step-wise Training Process

In order to observe the stability of performance, we provide the step-wise validating result. Figure 3 shows the training process of SwDA dataset. Intuitively, our multi-task model (blue line) outperforms than others at almost every step after 20000.

We take the best 20 values from the result of our multi-task (DA enhanced) training process and the ablative (no DA enhancement) training process, and make a significance test by Student's t-test. It comes to a result of $p = 7.18 \times 10^{-10}$, which suggests significant stable improvement of the supervision of our DA classification task comparing to the ablation setting.

In our experiment of cross-corpus condition, we find that if λ is set to a greater value, the accuracy will converge faster, but it does not work much

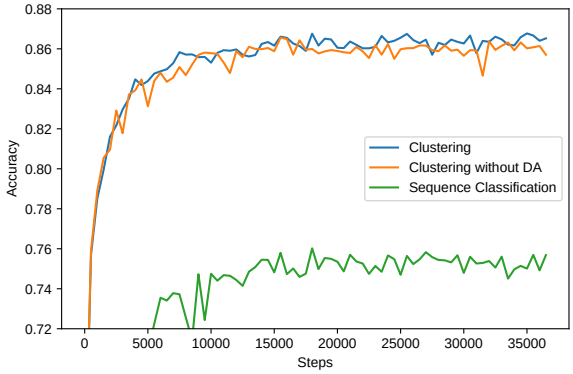

Figure 3: The ACC result of the SwDA validation set for every 500 training step.

at the highest point, which is mainly because the model is overfit on the auxiliary task. If λ is set to a suitable value, the regularization of cross-corpus multitasking will be more evident.

6 Conclusion and Future Work

We propose a text-based dialogue speaker clustering model. Based on the theory of the dialogue structure, the model takes the advantage of the semantic content and the communicative functions explicitly with the design of the BERT layer and the correlation matrix respectively. The model is also enhanced by the idea of cross-corpus supervision with the set-specific dialogue act classification auxiliary task design. It finally generates the cluster labels of speakers with spectral clustering. Our model outperforms the sequence classification baseline on every test, and outperforms the non-DA ablation on almost every test.

We have noticed that further pre-training the model on dialogue data may be helpful to extract better semantic embeddings, which we will examine in the future. The method of modeling the correlation of long-distance utterances also needs to be explored.

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.

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.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher 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.

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).

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.

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.

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.

Ryosuke Nakanishi, Koji Inoue, Shizuka Nakamura, Katsuya Takanashi, and Tatsuya Kawahara. 2019. Generating fillers based on dialog act pairs for smooth turn-taking by humanoid robot. In *9th International Workshop on Spoken Dialogue System Technology*, pages 91–101, Singapore. Springer Singapore.

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.

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.

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.

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

Iulian V Serban and Joelle Pineau. 2015. Text-based speaker identification for multi-participant opendomain dialogue systems. In *NIPS Workshop on Machine Learning for Spoken Language Understanding. Montreal, Canada*.

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*.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962v2*.

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

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

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

Xuejing Zhang, Xueqiang Lv, and Qiang Zhou. 2018b. 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.

Zhuosheng Zhang and Hai Zhao. 2021. Advances in multi-turn dialogue comprehension: A survey.