3 Model

The architecture of the proposed model is shown in Figure 2. We feed four types of information to the model: topic- and word-level data of the utterance (*content*), preceding interventionist verbal behavior (*context*) and prior MISC annotations of utterances (*MISC*). Particularly, we empirically extracted previous 5 utterances as context and 10 previous codes as MISC[2], where we set "unk" as the default.

Embeddings. We built two types of embeddings, word embedding and topic embedding. We created word embeddings from Googles pretrained Word2Vec (Mikolov et al., 2013) and created topic embeddings from a trained LDA (Blei et al., 2003) specific to the corpus. We treated each MISC as one document and trained an embedding model.

Unified Representation. We apply Bi-directional LSTM (Bi-LSTM) (Graves and Schmidhuber, 2005) on the inputs. Dropouts (Srivastava et al., 2014) are applied on the outputs of Bi-LSTM. We merge the outputs by concatenation and feed the outputs to the dense layer to learn a unified representation of the utterance.

Joint Learning. We apply domain adversarial training (Ganin et al., 2016) only on the topic inputs from learned topic representations. Our intuition is that the topic distributions across different stages of the MI session could track the variations of patients' intents. We empirically split the conversation into three time stages: Stage 1-3 (i.e, beginning, middle, and end). The goal of domain adversarial training is converted to a time stage prediction task, which aims to differentiate topic themes both locally and globally. We used one-hot encoding to represent labels of the prediction tasks. We deploy softmax functions for both time stage and intention predictions. We use categorical cross entropy to jointly optimize the training process of the two classification tasks: domain classification and patient intent classification.

4 Experiments

Each utterance is lowercased and tokenized by NLTK (Bird et al., 2009). We filter out the utterances that are shorter than 5 tokens and then remove punctuations. Finally, we obtain 22432 patient utterances. The dataset is stratified and split into training set (80%), validation set (10%) and testing set (10%), as shown in Table 2. We train our models on the training set and run grid search to find the optimal parameters on the validation set by the weighted F1 score.

Table 2: Statistics of the processed dataset.

Datasets	Train	Valid	Test	Total
CT	6246	779	768	7794
ST	3099	406	401	3906
FN	8600	1058	1074	10732
All				22432

The details of optimized parameters are listed as follows. The models were trained for 15 epochs with a batch size of 64. Each utterance and its context are padded to 50 words. The utterance's previous MISC codes are padded to 10. We pad the sequences with an "unknown"-token. The size of LSTMs was tuned in the range of $[100, 150, 200]$ and the size of dense layer tuned within $[100, 150, 200]$. We select the activation function of the Dense layer within {relu (Hahnloser et al., 2000), tanh, softplus} (Hahnloser et al., 2000). We tried different flip gradient value within $[0.05, 0.01, 0.005]$ for the domain adversarial training. We tuned the dropout rate between $[0.1, 0.2]$. The optimizer was selected either RMSprop (Hinton et al., 2012) or Adam (Kingma and Ba, 2014) with a fixed learning rate of 0.001. Finally, we empirically set the loss weight of the domain adversarial training to 0.05.

We trained the topic model on the MI corpus using Gensim (Řehůřek and Sojka, 2010). The number of topics was selected by coherence scores among 5, 10, 20 topics. We used Google pre-trained word embedding with 300 dimensions (Mikolov et al., 2013). We obtained 50-dimension code embedding by Word2vec (Mikolov et al., 2013) for the MISC codes, where each sequence of MISC were treated as a document.

We select three different approaches as our baselines with the inputs: content, context, MISC, and topic.

- (Pérez-Rosas et al., 2017) with rich linguistic features (denote as Perez2017_lin): We reproduced their method. We used scikit-learn (Buitinck et al., 2013) to ex-

[2]We encode 10 MISC codes prior to the current one as a sequence of 10 "words", then we treat the sequence as an additional input document.