中国科学院软件研究所学术年会’2021
暨计算机科学国家重点实验室开放周

学术论文

Dialogue Disentanglement in Software Engineering: How Far are We?

面向软件工程领域的对话解耦技术：我们还有多远？

江子攸，石琳，Celia Chen，胡军，王青

The 30th International Joint Conference on Artificial Intelligence (IJCAI-21)

主要联系人：石琳（15001193593，shilin@iscas.ac.cn）

Introduction

- Dialog disentanglement is a natural language task for disentangling valuable software chat messages into distinct conversations, which is an essential prerequisite for in-depth analyses that utilize this information. A number of approaches has been proposed to address such issue of dialog entanglement, such as message-pairs models (FF, CNN *etc.*) or sequential-based models (BERT, E2E *etc.*).
- Unlike general conversations, software engineering (SE) dialogs have different and distinct characteristics: (1) SE dialogs heavily focus on resolving issues, which are mostly in the form of question and answer; (2) SE dialogs are domain-specific and each domain has its own technical terms and concepts; (3) SE dialogs usually involve more complex problems, which require developers to discuss various topics within one dialog.
- We conduct an exploratory study on 7,226 real-world developers' dialogs mined from eight popular open-source projects hosted on online forum: Gitter. The main contributions are summarized: (1) We conduct a comparative empirical study on evaluating the state-of-the-art disentanglement approaches on software-related chat; (2) We propose a novel measure, DLD, for quantitatively measuring human satisfaction on disentangled results; (3) We release a dataset of disentangled software-related dialogs to facilitate the replication of our study and future improvements of disentanglement models.

Experiment

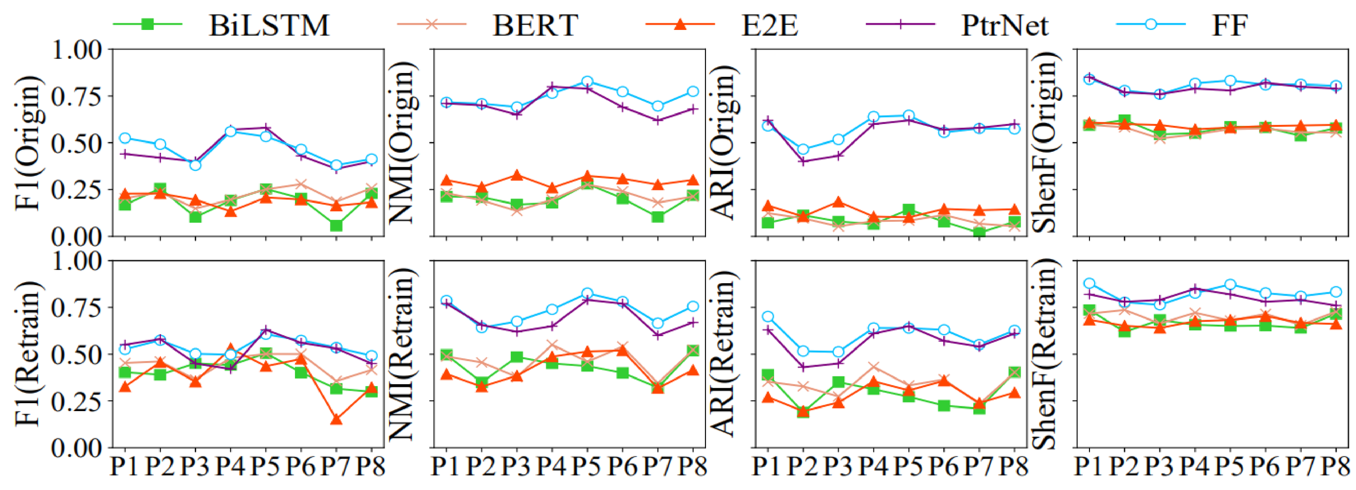
Id	Project	Domain	Entire Population		Sample Population		
			PA	UT	PA	DL	UT
P1	Angular	Frontend	22,467	695,183	125	97	778
P2	Appium	Mobile	3,979	29,039	73	87	724
P3	Dl4j	Data Science	8,310	252,846	93	100	1,130
P4	Docker	DevOps	8,810	22,367	74	90	1,126
P5	Ethereum	Blockchain	16,154	91,028	116	96	516
P6	Gitter	Collaboration	9,260	34,147	87	86	515
P7	Typescript	Language	8,310	196,513	110	95	1,700
P8	Nodejs	Web App	18,118	81,771	144	98	737
Total			*95,416*	*1,402,894*	*822*	*749*	*7,226*

Dataset is constructed from the most participated projects found in eight popular domains. The total number of participants is 95,416, accounting for 13% entire Gitter's participant population.

Model	Code	Dataset	Technology
Weighted-SP	No	No (Linux)	Weight Calculation
ME Classifier	No	No (Ubuntu)	Traditional Classifier
BiLSTM	Yes	Yes (Movie)	Recurrent NN
CISIR	No	Yes (News)	Convolutional NN
FF	Yes	Yes (Ubuntu)	FeedForward NN
BERT	Yes	Yes (Movie)	Encoder/Decoder NN
E2E	Yes	Yes (Movie)	Encoder/Decoder NN
PtrNet	Yes	Yes (Ubuntu)	Encoder/Decoder NN

	P	R	**F1**	loc_3	MAP	MRR	**NMI**	**ARI**	**ShenF**
Weighted-SP			√						√
ME Classifier	√	√							
BiLSTM			√		√		√	√	√
CISIR				√					√
FF	√	√	√	√		√			√
BERT			√				√	√	√
E2E			√				√	√	√
PtrNet	√	√	√				√	√	

- **Models Selection:** Search the literature published in the representative venues for the last 15 years.
- **Metrics Selection:** Investigate the evaluation measures that are adopted by existing literature.



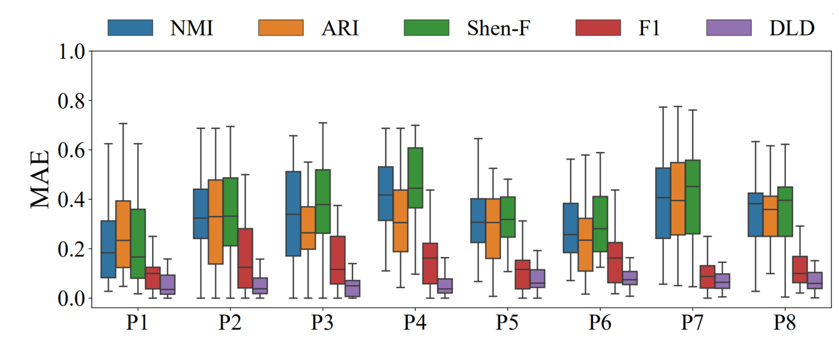
- **Experiment 1: Original:** Trained in the existing literature to disentangle our software-related chat.
- **Experiment 2: Retraining:** Retrain the five SOTA models on our software-related chat.

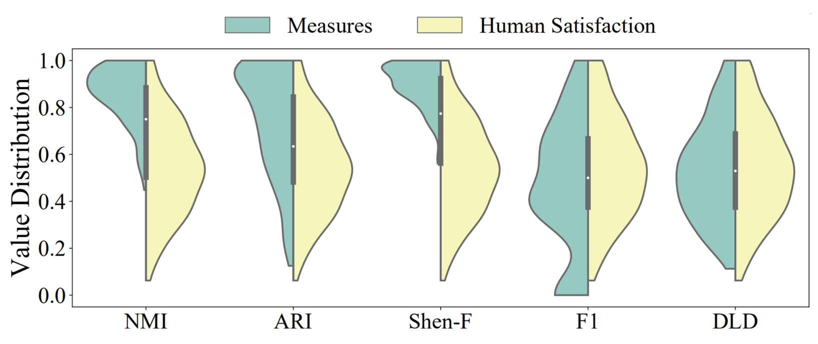
Measurement

A Novel Measure: *DLD*

- Dialog Levenshtein Revision:
$$DLR_v = E[\sigma(\Delta(D_T, D_P), \eta)]$$
- Dialog Levenshtein Ratio:
$$DLR_t = E[1 - \Delta(D_T, D_P)/(D_T + D_P)]$$
- Dialog Levenshtein Distance:
$$DLD = \lambda \cdot DLR_t + (1-\lambda) \cdot DLR_v, 0 \le \lambda \le 1$$

Effectiveness of *DLD*





Category	Error		Correlation	Hypothesis		
Analysis	RMSE	MAE	PEA	IST	PST	ANOVA
ME Classifier	0.38	0.34	0.08	E-55	E-46	E-55
BiLSTM	0.37	0.32	0.02	E-19	E-17	E-19
CISIR	0.41	0.36	0.17	E-69	E-59	E-69
FF	0.19	0.14	0.85	E-4	E-14	E-4
BERT	**0.08**	**0.92**	**0.92**	0.51	0.31	0.51

- The lowest error (RMSE: 0.08, MAE: 0.07), highest correlation (PEA: 0.92).
- No significant differences when compared to human satisfaction (Hypothesis).
- The lowest error across all projects (Figure).

Bad Cases

Interaction Pattern

Bad Case 1: Ignoring Interaction Patterns (IIP: 64%)

Miss
R_1: Does this approach make any sense? $<OQ, R_i>$
R_2: If you want to leverage caching of build tasks, yes. $<PA, R_s>$
R_1: Copy what I need into a docker image? $<FQ, R_i>$
R_2: You get my point! $<FD, R_s>$

Contextual-related: data model & mongodb

Bad Case 2: Ignoring Contextual Information (ICI: 21%)

Miss
R_1: Can it be represented in **data models**?
R_2: That's exactly why we have **mongodb**.

Bad Case 3: Mixing up Topics (MT: 9%)

R_1: How can I get it back please?
R_2: Try to install EasyDex.
R_1: How can I see my outstanding balance?
R_2: Use EasyDex its light wallet.
R_1: Thanks bro.

Relation (P1, P3) = "friend"

Bad Case 4: Ignoring User Relationships (IUR: 6%)

Miss
R_1: Should I give up applying angular?
R_2: Why give up?
R_3: Igor will find and kill u :)

Conclusion

- We evaluate five SOTA dialog disentanglement models on SE dialogs to investigate how these models can be used in the context of SE.
- We conduct two experiments with the original and the retrained models respectively. Results show that the original FF model is the best one for disentangling SE dialogs.
- We introduce a novel measure DLD. Compared to other measures, DLD can more accurately measure human satisfaction.
- We investigate the reasons why some disentangled dialogs are unsatisfying, and identify four common bad cases.