対話破綻検出チャレンジ2

The Dialogue Breakdown Detection Challenge 2

東中竜一郎[1*]　船越孝太郎[2]　稲葉通将[3]　荒瀬由紀[4]　角森唯子[5]
Ryuichiro Higashinaka[1]　Kotaro Funakoshi[2]　Michimasa Inaba[3]
Yuki Arase[4]　Yuiko Tsunomori[5]

[1] NTT メディアインテリジェンス研究所　NTT Media Intelligence Laboratories
[2] (株) ホンダ・リサーチ・インスティチュート・ジャパン　Honda Research Institute Japan Co., Ltd.
[3] 広島市立大学　Hiroshima City University
[4] 大阪大学　Osaka University
[5] NTT ドコモ　NTT DoCoMo

Abstract: Detecting breakdowns in dialogue can be one of the promising techniques in dialogue systems. To propel the research and development of dialogue breakdown detection, we have been organizing the dialogue breakdown detection challenge. This report describes the second edition of the challenge: dialogue breakdown detection challenge 2 (DBDC2). One notable difference from the last year's challenge is that we utilized dialogue logs of three systems instead of one in order to examine the generality of dialogue breakdown detection technology. This report provides the overview of DBDC2 and describes the details of the datasets, evaluation metrics, dialogue systems used, and the results of the submitted runs by eight participating teams.

1　はじめに

昨年より，対話破綻検出チャレンジと題し，人間と対話システムとの間で生じる「対話破綻」（ユーザが対話を継続できなくなる状態）を自動検出することを目的とした，評価型ワークショップを開催している．

対話破綻検出技術が実現すれば，システムが自分の発話をする前に，その発話が対話破綻に繋がることが分かり，対話破綻を事前に回避することができるし，また，たとえ対話破綻してしまったとしても，破綻したことが認識できれば，リカバリを試みることも可能となる．対話システムが今後普及していくためには，少なくとも対話が破綻しないことは大前提であろう．対話破綻検出技術の確立は今後の対話システムの普及のために極めて重要である．

本稿では，二回目のチャレンジである対話破綻検出チャレンジ2（Dialogue Breakdown Detection Challenge 2; DBDC2）について報告する．DBDC2 のタスク設定は DBDC1 と同様である．また，対象とする対話システムは，非タスク指向型対話システム（雑談対話システム）であり，こちらも昨年と同じである．ただし，DBDC2 は，前回のチャレンジ（DBDC1）から以下の点が異なる．

- DBDC1 では，1 つの対話システムのログを対象として対話破綻検出を行ったが，提案された対話破綻検出技術が他の対話システムについても効果があるかは不明であった．よって，汎用的な対話破綻検出技の確立のため，DBDC2 では，3 つの対話システムを対象として対話破綻検出を行う．
- DBDC1 では，ラベル一致系統と分布距離系統の評価尺度を用いたが，どの尺度を重視すればよいか分かりづらいという問題があった．DBDC2 では，分布距離系統（複数のアノテータが付与する対話破綻ラベルの分布の近さに基づく評価尺度）を重視する．これは，ラベルを一つに決めることがそもそも難しいこと，および，分布距離系統の方が，複数のアノテータの総意を反映した評価ができると考えられることによる．

DBDC2 では，昨年度を上回る 8 チームが参加し，さまざまな対話破綻検出手法が検討された．本稿では，タスク設定，対象とした対話システム・配布データ，評価尺度，各チームの手法概要およびその性能比較を行い，その結果得られた知見や今後の展開について述べる．なお，DBDC1 については，文献 [1, 2, 3] に詳細があるので参照されたい．

2　タスク設定

対話破綻検出チャレンジのタスクは，ある対話文脈について後続するシステム発話があるとき，そのシステム発話が対話破綻（対話破綻の類型等については文献 [4] を参照）を引き起こすかを検出することである．

本チャレンジでは，同じ発話について多数の評価者

*連絡先：higashinaka.ryuichiro@lab.ntt.co.jp

（アノテータ）を用いて対話破綻のラベル付けを行い，ラベルの分布によって対話破綻らしさを表現する．参加者は，対話破綻検出対象の各システム発話について，(1) 破綻かどうかのラベルと（2）破綻らしさの分布（○，△，×の確率分布）を提出する．そして，対話破綻検出器の評価は，多数の評価者が対話破綻とした箇所を正しく検出できるか（ラベル一致系統），実際のラベル分布を正しく推定できるか（分布距離系統）の二種類の観点から行う．DBDC2 では，DBDC1 と同様，対象発話より後の情報は対話破綻検出に用いない．また，フォーマルランでは，一つの参加チームについて 3 つまで結果（run と呼ぶ）を提出することができる．

3 配布データ

開発・評価用に新たに対話データを収集し，破綻アノテーションを行って配布した．データ収集およびアノテーションの方法は DBDC1 と同じであるが，前述したように，より汎用的な対話破綻検出技術の実現のため，DBDC2 では以下の 3 つの対話システムを用いて対話データを収集した．

DCM NTT ドコモが一般公開している雑談対話 API[1] を用いた雑談対話システム [5].

DIT デンソーアイティーラボラトリ提供の雑談対話システム [6, 7]．本システムでは，まず対話コーパスから，キーワードと対話パターン (キーワードをその型を表すスロットで置き換えたもの) との相関を表す 2 部グラフを抽出し，そのグラフ上のラプラシアンラベル伝搬により, 与えられたキーワードと関連性の高い他のキーワードと対話パターンを見つける．そして，これに入力発話中のキーワードを当てはめ, 得られるキーワードと対話パターンから意味的な繋がりのある応答候補を生成する．

IRS オーガナイザが準備した，IR-STATUS[8] に準拠した用例ベースの雑談対話システム．ユーザ発話に対し，最も類似した入力部を持つ用例を検索し，その応答部を用いて発話する．用例間の類似度の計算には，全文検索ライブラリ Apache Lucene をそのまま用いている．形態素解析器には Lucene に付属している JapaneseAnalyzer(Kuromoji) を用い，使用した用例は 26972 個である．

対話データ収集に際しては，これらのシステムと対話ができるウェブサイトを構築し，対話参加者にウェブブラウザでサイトにアクセス・対話してもらって収集した．収集された対話数は，各システムにつき 100 対話で，全部で 300 対話である．各対話はシステムのプロンプトで始まり，その後ユーザとシステムが交互にそれぞれ 10 回発言したところで強制的に終了となる．このため，一対話は 21 発話からなる．

収集された対話データへのアノテーションについては，

○ **破綻ではない** 当該システム発話のあと対話を問題無く継続できる．

△ **破綻と言い切れないが，違和感を感じる発話** 当該システム発話のあと対話をスムーズに継続することが困難．

× **あきらかにおかしいと思う発話．破綻** 当該システム発話のあと対話を継続することが困難．

の三種類のラベルを，システム発話毎に 30 名のアノテータが付与した．なお，○，△，×の記号は，配布データ中ではそれぞれ O，T (Triangle の意)，X のアルファベットにより表現されている．本稿ではこれらの表記の両方を用いるため注意されたい．

表 1 に学習，開発，テストデータに関する統計量をまとめた．表中の「アノテータ数」は各対話に対するアノテータの数であり，延べ数ではない．

各セッションにおける単語数，語彙数の平均をシステム・ユーザそれぞれについて求めた結果を表 2 に示す． 形態素解析には Mecab[2] を利用した．

DBDC1 では，DCM の対話のうち，20 対話を学習用データ，80 対話を評価用データとして配布した．一方，DBDC2 では，DCM/DIT/IRS による対話の半分（50 対話）を学習用として配布し，残りの 50 対話を評価用として配布した．これは，DCM については雑談対話コーパス [3] や昨年度の開発・評価用データが利用できるものの，DIT，IRS のシステムについては参加者が利用できるデータが事前になく，20 対話ではチューニングなどの目的に不十分と考えられたからである．

4 評価方法

ラベル一致系統と分布距離系統の二つの系統の評価尺度を用いる．これらの評価尺度は DBDC1 と同様のため，ここでは簡単に述べるにとどめる．詳細については [1] を参照されたい．

ラベル一致系統 各システム発話について，付与された対話破綻ラベルの多数決を取り，該システム発話が破綻かどうかのラベルを一つに決め，評価データと検出器のラベルを比較することで評価を行う．評価尺度として，Accuracy（X を破綻とみなした場合の正解率），破綻ラベル X の Precision, Recall, F-measure，破綻ラベル T と X を合わせて破綻とした場合の Precision, Recall, F-measure

[1] https://www.nttdocomo.co.jp/service/developer/smart_phone/analysis/chat/

[2] http://taku910.github.io/mecab/

[3] https://sites.google.com/site/dialoguebreakdowndetection/chat-dialogue-corpus

表 1: 学習/開発/テストデータの統計

	雑談対話コーパス		DBDC1	DBDC2		
	init100	rest1046	開発/評価	DCM (開発/評価)	DIT (開発/評価)	IRS (開発/評価)
対話数	100	1046	20/80	50/50	50/50	50/50
アノテータ数	24	2 or 3	30	30	30	30
O	59.2%	58.3%	37.1%	39.8%	33.0%	37.4%
T	22.2%	25.3%	32.2%	30.2%	27.4%	24.3%
X	18.6%	16.4%	30.6%	29.9%	39.5%	38.3%
Fleiss' κ	0.28	0.28*	0.20	0.31	0.24	0.36
κ (T=X)**	0.40	0.40*	0.27	0.44	0.38	0.48

* セット毎の κ 値のマクロ平均，**T を X と見なした場合

表 2: セッションごとの単語数・語彙数の統計（95% 信頼区間）

	DCM		DIT		IRS	
	システム	ユーザ	システム	ユーザ	システム	ユーザ
単語数	82.7 ± 4.0	80.5 ± 5.4	319.4 ± 11.6	103.3 ± 7.2	152.9 ± 6.4	98.4 ± 6.8
語彙数	40.6 ± 1.9	43.7 ± 2.2	121.4 ± 3.8	54.5 ± 2.9	82.0 ± 2.9	53.1 ± 2.9

の 7 つがある．これらの尺度は，DBDC2 では補助的な指標として位置づける．なお，適切な値の設定が難しいことから，DBDC1 で用いていたパラメータ t を廃止し，純粋な多数決とする．ラベルの数が同数の場合，○，△，×の順の優先度で正解ラベルを決定する．

分布距離系統 対話破綻ラベルの分布に基づく評価尺度として，Jensen-Shannon Divergence による分布間の距離と分布間の平均二乗誤差の二種類を用いる．それぞれについて，ラベルの分布をそのまま用いる場合，ラベル T と X を同一の破綻ラベルとみなした場合，ラベル O と T を同一の破綻ラベルとみなした場合について算出するため，評価尺度は全部で 6 つある．これらの尺度は，DBDC2 ではプライマリな指標として位置付ける．

5 手法概要と結果

ここでは，オーガナイザが用意したベースライン手法と 8 チームが考案した手法およびその結果について概観する．各チームの手法の詳細については，本予稿集に含まれる，それぞれの報告を参照されたい．

5.1 ベースライン

DBDC1 で用いた CRF によるベースライン [1]（以降，baseline）の他に，学習データ中の最頻ラベルを常に答えるベースライン（以降，majority），および 1/3 の確率でランダムにラベルを決定するベースライン（以降，random）を新たに導入した．ラベルの確率分布については，baseline は推定したラベルの確率を 1 とし，他は 0 とする．majority は学習データ中のラベル分布で，random は全て 1/3 で答える．

DCM については，baseline1 と baseline2 を用意した．baseline1 は DBDC2 の開発データのみで学習したもので，baseline1 は雑談対話コーパスと DBDC2 の開発データで学習したものである．

5.2 参加チームの手法概要

表 3 は参加チームの手法をまとめたものである（順番は結果の提出順）．DBDC1 同様，さまざまな対話破綻検出手法が試みられていることが分かる．DBDC1 では 6 チーム中 4 チームが深層学習に基づく手法であったが，今回は 8 チーム中 2 チームが破綻検出自体に深層学習を，4 チームが特徴量抽出にニューラルネットワークに基づく手法を用いている．

5.3 結果：ラベル一致系統

ラベル一致系統の評価尺度に関する各チームの結果を表 4 に示す．各尺度やシステムによって，最大性能を示すチームや run は異なるものの，全体の傾向として NTTCS（その中でも run2）が最も安定して高い性能を示しているようである．特に，一致率と F 値 (X) については，3 システム全てで NTTCS が最高性能を発揮している．T+X の F 値に絞れば，smap および RSL16BD が最高性能を示している．

DBDC1 に対する DBDC2 の特徴として，RSL16BD, NTTCS のように破綻のパターンによる場合分けを行っているシステムが高い性能を示している点が挙げられる．また完全に人手によるルールに基づく OKSAT も，最高点を示すことはないものの，肉薄する性能を示している点は興味深い．

5.4 結果：分布距離系統

分布距離系統の評価尺度に関する各チームの結果を表 5 に示す．分布距離系統では数値が小さいほど性能がよい．分布距離系統では，ラベル一致系統ほどのばらつきは見られず，一貫して NTTCS が高い性能を示している．ただし，HCU，smap，RSL16BD，KIT もよい性能で追随しており，それほど大きな差が見られるわけではない．

表 3: 参加チームの手法概要

チーム（組織）	用いた手法	特徴	各 run の違い
HCU（広島市立大）	RNN, MLP	RNN による特徴量生成と多層パーセプトロンによるラベリング．独自データも学習に利用．	run1: MSE 最少化， run2: F 値最大化, run3: MSE 最小化により 4 モデルを学習し平均を利用
Mtkn（はこだて未来大）	Doc2Vec, RF	Doc2Vec を用いて特徴量作成，Random Forest でラベリング．	run1 のみ．
smap（TIS）	NCM, SVM	Neural Conversational Model の出力を特徴量とし，SVM でラベリング．	run1: Encoder/Decoder 双方の出力を用いて学習．run2: Decoder の出力を利用．
RSL16BD（早稲田大）	Word2Vec	Word2Vec を用いて作成した特徴ベクトルと開発データの類似度を測定，開発データでの破綻確率を算出して用いる．また発話をパターン分けし，パターンごとに破綻確率を算出．	run1: 開発データの破綻確率を利用, run2: パターンごとの破綻確率を利用, run3: run1 と 2 の組み合わせ．
NTTCS（NTT）	破綻箇所・評価分布分析, ETR	破綻のパターンを抽出，パターンごとに特徴量を設計．Extra Trees Regression により分布推定．	run1〜run3: 特徴量，学習データを変化
KIT16 (京都工繊大)	MLP, LSTM, RCNN	Project Next NLP によって策定された破綻類型ごとに破綻検出器を学習，多層パーセプトロンによるラベリング．Google N-gram の利用．	run1: 破綻類型の利用，run2，run3 は破綻類型を利用せず．
OKSAT（大阪教育大）	ルール	破綻するケースを観察し，ルールを作成	run1〜run3: ルールを変化
kanolab（静岡大）	Word2Vec, ルール	Word2Vec による単語間距離の利用，疑問文に対して疑問文を返すケースの検出．BCCWJ も学習に利用．	run1: ユーザの話題転換後に古い話題のキーワードを用いているかをルールに追加，run2: run1 のルールを用いない, run3: 判定閾値を緩める．

6 まとめと今後の課題

昨年に引き続き，雑談対話システムにおける対話破綻を自動検出することを目的とした「対話破綻検出チャレンジ 2」を開催し，タスク設定，対話システムと配布データ，および，各参加チームの手法と結果について概観した．

今回参加した 8 チームの中では，NTTCS のものが比較的良好な結果を示しており，対話破綻のパターンを抽出・パターンごとに特徴量を設計することが有効である可能性が示された．また，今回プライマリな尺度として採用した分布距離系統でよい成績を収めたものには，深層学習を用いたものが含まれており，これらの技術の進展により，さらなる対話破綻検出の精度向上も期待できる．

今回 3 つの対話システムを用いたが，このことにより，現状の対話破綻検出器がどの程度の汎用性を持っているかについても把握することができるようになったと言える．詳細な分析はこれから行っていかなくてはならないが，今後に向けた有益な知見が得られたと考えている．来年も対話破綻検出チャレンジ 3 を実施し，さらなる技術の発展を目指したい．また，雑談対話だけではなくタスク指向型対話，テキスト対話だけではなくマルチモーダル対話，についても同様のチャレンジを実施できればと考えている．

謝辞

開発・評価用データの作成にあたっては人工知能学会より特別補助をいただきました．対話データ収集において，NTT ドコモの雑談対話 API を使わせていただきました．また，DIT システムは，株式会社デンソーアイティーラボラトリの塚原裕史様，内海慶様にご提供いただきました．感謝いたします．加えて，タイトなスケジュールにもかかわらず参加いただいた参加チームの皆様にも感謝いたします．

参考文献

[1] 東中, 船越, 小林, 稲葉. 対話破綻検出チャレンジ. 第 75 回言語・音声理解と対話処理研究会 (第 6 回対話システムシンポジウム), 人工知能学会研究会資料 SIG-SLUD-75-B502, pp. 27–32, 2015.

[2] R. Higashinaka, K. Funakoshi, Y. Kobayashi, and M. Inaba. The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics. In *Proc. LREC*, 2016.

[3] K. Funakoshi, R. Higashinaka, M. Inaba, Y. Kobayashi, S. Sugawara, K. Takanashi, H. Otsuka, H. Koiso, and M. Bono. On dialogue breakdown: Annotation and detection – A report from the dialogue breakdown detection challenge. In *Proc. WOCHAT*, 2016.

[4] 東中, 船越, 荒木, 塚原, 小林, 水上. テキストチャットを用いた雑談対話コーパスの構築と対話破綻の分析. 自然言語処理, Vol. 23, No. 1, 2016.

[5] 大西, 吉村. コンピュータとの自然な会話を実現する雑談対話技術. NTT DoCoMo テクニカル・ジャーナル, Vol. 21, No. 4, pp. 17–21, 2014.

[6] H. Tsukahara and K. Uchiumi. System utterane generation by label propagation over association graph of words and utterance patterns for open-domain dialogue systems. In *Proc. PACLIC*, 2015.

[7] 塚原, 内海. 対話行為と話題推定によるラベル伝搬を利用した雑談生成方法の改良. 第 30 回人工知能学会年次大会, 2016.

[8] A. Ritter, C. Cherry, and W. B. Dolan. Data-driven response generation in social media. In *Proc. EMNLP*, pp. 583–593, 2011.

表 4: ラベル一致系統の評価尺度による各チームの結果

system	team	run	Accuracy	Precision (X)	Recall (X)	F-measure (X)	Precision (T+X)	Recall (T+X)	F-measure (T+X)
DCM	HCU	run1	0.496	**0.516**	0.371	0.431	0.910	0.340	0.495
		run2	0.484	0.486	0.298	0.369	0.859	0.577	0.690
		run3	0.504	**0.520**	0.292	0.374	0.910	0.396	0.551
	Mtkn	run1	0.416	0.373	0.230	0.285	0.736	0.226	0.345
	smap	run1	0.415	0.434	0.478	0.455	0.739	**0.908**	**0.815**
		run2	0.415	0.372	0.393	0.383	0.741	0.526	0.616
	RSL16BD	run1	0.405	0.000	0.000	0.000	**1.000**	0.003	0.006
		run2	0.507	**0.527**	0.438	0.479	0.892	0.368	0.521
		run3	0.495	**0.527**	0.438	0.479	0.845	0.426	0.567
	NTTCS	run1	0.527	0.477	0.652	**0.551**	0.842	0.710	**0.770**
		run2	**0.565**	**0.523**	0.584	**0.552**	0.875	0.624	0.728
		run3	0.520	0.478	0.657	**0.553**	0.830	0.705	0.762
	KIT16	run1	0.476	0.423	0.663	0.516	0.768	0.599	0.673
		run2	0.455	0.453	0.573	0.506	0.753	0.825	**0.787**
		run3	0.462	0.500	0.309	0.382	0.810	0.604	0.692
	OKSAT	run1	0.473	0.411	0.736	**0.527**	0.777	0.691	0.732
		run2	0.427	0.446	0.185	0.262	0.770	0.159	0.263
		run3	0.355	0.382	0.854	**0.528**	0.736	0.816	**0.774**
	kanolab	run1	0.433	0.372	**0.927**	**0.531**	0.727	**0.900**	**0.804**
		run2	0.424	0.364	0.770	0.495	0.731	0.766	0.748
		run3	0.407	0.350	0.629	0.450	0.716	0.638	0.675
	random		0.338	0.316	0.337	0.326	0.650	0.688	0.668
	majority		0.405	0.000	0.000	0.000	0.000	0.000	0.000
	baseline1		0.460	0.407	0.787	**0.536**	0.746	0.794	0.769
	baseline2		0.451	**0.507**	0.191	0.278	0.811	0.574	0.672
DIT	HCU	run1	**0.622**	**0.652**	0.814	**0.724**	0.901	0.748	0.817
		run2	0.569	**0.643**	0.777	0.703	**0.875**	0.920	**0.897**
		run3	**0.624**	**0.655**	0.818	**0.727**	**0.904**	0.842	**0.872**
	Mtkn	run1	0.382	0.500	0.265	0.347	0.757	0.257	0.384
	smap	run1	0.584	0.597	0.818	0.690	0.847	**0.993**	**0.914**
		run2	0.545	0.566	0.731	0.638	**0.863**	0.782	0.820
	RSL16BD	run1	0.591	0.540	**1.000**	0.701	0.843	**1.000**	**0.915**
		run2	0.602	0.557	**0.996**	0.715	0.852	**0.976**	**0.910**
		run3	0.602	0.557	**0.996**	0.715	0.852	**0.976**	**0.910**
	NTTCS	run1	**0.640**	0.615	0.939	**0.744**	**0.895**	0.910	**0.903**
		run2	**0.655**	**0.632**	0.943	**0.757**	**0.900**	0.891	**0.895**
		run3	**0.644**	0.617	0.939	0.745	**0.895**	0.913	**0.904**
	KIT16	run1	0.525	0.533	0.803	0.640	0.817	0.789	0.802
		run2	0.591	0.594	0.837	0.695	**0.865**	0.932	**0.897**
		run3	0.553	**0.623**	0.652	0.637	**0.880**	0.709	0.785
	OKSAT	run1	0.589	0.542	**0.992**	0.701	0.843	**0.988**	**0.909**
		run2	0.551	0.536	0.898	0.671	0.835	0.896	0.864
		run3	0.487	0.543	**0.951**	0.691	0.840	0.942	**0.888**
	kanolab	run1	0.571	0.532	**0.989**	0.691	0.823	**0.981**	**0.895**
		run2	0.553	0.536	0.902	0.672	0.827	0.891	0.857
		run3	0.518	0.523	0.765	0.622	0.821	0.769	0.794
	random		0.320	0.439	0.299	0.356	0.760	0.677	0.716
	majority		0.480	0.480	**1.000**	0.649	0.749	**1.000**	0.857
	baseline		0.591	0.555	**0.955**	0.702	**0.850**	**0.966**	**0.905**
IRS	HCU	run1	0.531	**0.567**	0.589	0.577	**0.777**	0.538	0.636
		run2	0.482	0.527	0.602	0.562	0.741	0.779	0.760
		run3	0.505	0.534	0.580	0.556	0.757	0.602	0.671
	Mtkn	run1	0.396	0.402	0.203	0.270	0.650	0.213	0.321
	smap	run1	0.420	0.483	0.541	0.510	0.725	**0.983**	**0.835**
		run2	0.464	0.489	0.563	0.523	0.721	0.653	0.685
	RSL16BD	run1	0.393	0.000	0.000	0.000	0.000	0.000	0.000
		run2	0.551	0.494	**0.961**	**0.653**	0.739	0.930	**0.824**
		run3	0.553	0.497	**0.961**	**0.655**	0.740	0.927	**0.823**
	NTTCS	run1	**0.578**	0.537	0.823	**0.650**	**0.785**	0.810	**0.797**
		run2	**0.584**	**0.554**	0.801	**0.655**	**0.791**	0.773	0.782
		run3	**0.584**	**0.539**	0.840	**0.657**	**0.789**	0.826	**0.807**
	KIT16	run1	0.520	0.492	0.762	0.598	0.743	0.745	0.744
		run2	0.498	0.506	0.688	0.583	**0.750**	0.857	**0.800**
		run3	0.455	0.505	0.442	0.471	0.748	0.647	0.694
	OKSAT	run1	0.531	0.487	0.883	**0.628**	0.740	0.868	**0.799**
		run2	0.465	0.510	0.429	0.466	0.722	0.392	0.508
		run3	0.451	0.501	0.861	**0.634**	**0.756**	0.840	**0.796**
	kanolab	run1	0.480	0.457	0.788	0.579	0.714	0.796	0.752
		run2	0.482	0.461	0.766	0.576	0.716	0.770	0.742
		run3	0.469	0.454	0.658	0.537	0.725	0.681	0.702
	random		0.338	0.389	0.312	0.346	0.663	0.683	0.673
	majority		0.420	0.420	**1.000**	0.592	0.649	**1.000**	0.787
	baseline		0.536	0.492	0.818	0.615	**0.762**	0.852	**0.804**

各尺度で，システム別に最大値を下線を引いた．また同範囲内で最大値の 95%に入る数値を太字にした．

表 5: 分布距離系統の評価尺度による各チームの結果

system	team	run	JS (O,T,X)	JS (O,T+X)	JS (O+T,X)	MSE (O,T,X)	MSE (O+T,X)	MSE (O+T,X)
	HCU	run1	0.101	0.073	0.062	0.056	0.075	0.067
		run2	0.118	0.078	0.073	0.068	0.084	0.078
		run3	0.100	0.072	0.061	0.055	0.074	0.067
	Mtkn	run1	0.480	0.426	0.260	0.257	0.415	0.231
	smap	run1	0.146	0.104	0.093	0.070	0.099	0.092
		run2	0.182	0.133	0.120	0.099	0.138	0.134
	RSL16BD	run1	0.142	0.112	0.081	0.072	0.107	0.077
		run2	0.095	0.064	0.061	0.049	0.064	0.060
		run3	0.094	0.064	0.060	0.049	0.064	0.060
	NTTCS	run1	0.087	0.057	0.056	0.045	0.057	0.055
		run2	**0.085**	0.057	**0.054**	**0.044**	**0.056**	**0.054**
		run3	0.086	**0.056**	0.056	0.045	**0.056**	0.056
DCM	KIT16	run1	0.102	0.069	0.066	0.053	0.070	0.065
		run2	0.119	0.076	0.078	0.066	0.082	0.080
		run3	0.120	0.078	0.077	0.066	0.083	0.080
	OKSAT	run1	0.423	0.256	0.327	0.220	0.231	0.305
		run2	0.467	0.429	0.232	0.248	0.421	0.200
		run3	0.499	0.286	0.441	0.269	0.257	0.425
	kanolab	run1	0.464	0.222	0.437	0.247	0.191	0.425
		run2	0.468	0.261	0.402	0.249	0.234	0.388
		run3	0.481	0.303	0.383	0.257	0.280	0.367
	random		0.145	0.111	0.085	0.074	0.111	0.077
	majority		0.143	0.112	0.083	0.072	0.108	0.076
	baseline1		0.432	0.237	0.350	0.227	0.209	0.328
	baseline2		0.410	0.279	0.215	0.223	0.256	0.182
	HCU	run1	0.055	0.036	0.038	0.031	0.035	0.044
		run2	0.078	0.045	0.056	0.045	0.045	0.065
		run3	0.052	0.033	0.035	0.029	0.032	0.041
	Mtkn	run1	0.499	0.444	0.305	0.266	0.437	0.270
	smap	run1	0.095	0.071	0.060	0.042	0.061	0.060
		run2	0.118	0.086	0.079	0.064	0.084	0.096
	RSL16BD	run1	0.106	0.086	0.064	0.055	0.081	0.063
		run2	0.053	0.034	0.037	0.030	0.035	0.041
		run3	0.052	0.033	0.036	0.029	0.034	0.040
	NTTCS	run1	0.047	**0.030**	0.031	**0.025**	**0.029**	**0.034**
		run2	**0.046**	**0.030**	**0.030**	**0.025**	**0.029**	**0.034**
		run3	0.047	**0.030**	0.031	**0.025**	**0.029**	0.035
DIT	KIT16	run1	0.076	0.054	0.052	0.043	0.055	0.060
		run2	0.064	0.042	0.042	0.037	0.044	0.050
		run3	0.066	0.044	0.045	0.037	0.044	0.053
	OKSAT	run1	0.354	0.148	0.350	0.173	0.104	0.323
		run2	0.376	0.186	0.348	0.187	0.148	0.321
		run3	0.410	0.213	0.394	0.210	0.171	0.367
	kanolab	run1	0.372	0.163	0.368	0.185	0.121	0.342
		run2	0.384	0.194	0.354	0.192	0.157	0.326
		run3	0.403	0.239	0.343	0.205	0.208	0.314
	random		0.114	0.091	0.067	0.061	0.089	0.069
	majority		0.111	0.091	0.067	0.059	0.089	0.066
	baseline		0.354	0.153	0.334	0.173	0.109	0.305
	HCU	run1	0.118	0.080	0.082	0.065	0.079	0.091
		run2	0.156	0.100	0.114	0.091	0.102	0.128
		run3	0.118	0.081	0.082	0.066	0.080	0.091
	Mtkn	run1	0.515	0.468	0.323	0.283	0.460	0.300
	smap	run1	0.151	0.111	0.102	0.073	0.099	0.104
		run2	0.186	0.135	0.130	0.102	0.136	0.148
	RSL16BD	run1	0.158	0.124	0.100	0.082	0.115	0.101
		run2	0.109	0.073	0.076	0.058	0.070	0.081
		run3	0.110	0.074	0.076	0.059	0.071	0.081
	NTTCS	run1	**0.099**	**0.066**	**0.069**	**0.053**	**0.064**	**0.073**
		run2	0.101	0.068	0.070	0.054	0.065	0.074
		run3	**0.099**	**0.066**	**0.069**	**0.053**	**0.064**	**0.073**
IRS	KIT16	run1	0.117	0.078	0.083	0.064	0.077	0.090
		run2	0.123	0.079	0.087	0.070	0.081	0.095
		run3	0.131	0.085	0.092	0.074	0.085	0.102
	OKSAT	run1	0.393	0.217	0.353	0.205	0.191	0.334
		run2	0.456	0.379	0.305	0.246	0.366	0.281
		run3	0.428	0.266	0.381	0.231	0.239	0.361
	kanolab	run1	0.426	0.256	0.375	0.228	0.232	0.359
		run2	0.424	0.261	0.366	0.226	0.238	0.349
		run3	0.434	0.288	0.356	0.232	0.267	0.339
	random		0.160	0.121	0.101	0.085	0.115	0.102
	majority		0.157	0.122	0.101	0.081	0.114	0.100
	baseline		0.386	0.215	0.339	0.203	0.188	0.319

各尺度で，システム別に最小値を太字にした．