

tracrin: コミュニケーション研究で用いられる定性的記述「トランスクリプト」作成支援プログラム

tracrin: Program to support creating the qualitative descriptions used in communication research

牧野 遼作^{1,2}, 栗原 勇人², 谷貝 祐介², 門田 圭祐², 臼田 泰如³
Ryosaku Makino, Yuto Kurihara, Yusuke Yagai, Keisuke Kadota, Yasuyuki Usuda

¹広島工業大学, ²早稲田大学, ³ 国立国語研究所

Hiroshima Institute of Technology, Waseda University, National Institute for Japanese Language and Linguistics
rmakino19@gmail.com

概要

トランスクリプトは、人々のコミュニケーションを収録・録音し、定性的に分析するために欠かせない研究資源である。トランスクリプトの作成には膨大な時間を要するが、発表媒体などにあわせて調整するなど、研究にとって本質的ではない作業も多く含まれる。本発表で報告する tracrin は、発話の重なりインデント位置の自動調整を行うシステムである。本システムは、単に便利なツールだけでなく、定性的研究資源を利用した新たなデータベース研究の第一歩となりうるものである。

キーワード：会話分析(conversation analysis), 書き起こし(transcript), 会話データベース(conversation database)

1. はじめに

本稿は、社会学の会話分析研究など、人々のコミュニケーションをビデオ収録し、定性的な手法を用いてアプローチする際に作成・利用されるトランスクリプト(書き起こし)の作成支援プログラム“tracrin”の概要、利用方法、内部仕様について主に報告を行う。

現在人々がコミュニケーションする姿を収録し、その中でなされる発話から身体的な振る舞いまでを対象とした研究は、言語学、心理学、教育学、社会学、認知科学と数多くの領域でなされている。中でも録音・収録した音声/ビデオデータは、繰り返し観察することができる。この観察に基づいた定性的アプローチをもとにした研究も、談話分析[1], 文脈分析[2], 会話分析[3]などと数多く存在する。

このような音声/ビデオ観察に基づく人々のコミュニケーションに対する定性的アプローチ(以下、定性的アプローチ)では、“自身以外の研究者や読者”(以下、読者)に、観察したデータの概要をエビデンスとして示さなければならない。現在、多くのアプローチで作成・利用されるのがトランスクリプト(書き起こし)という方法である。トランスクリプトでは、データ内の発話や身体的振る舞いを時間順に書き起こすことで、データ

内の人々の振る舞いを読者に提示し、その順番やタイミングを論拠に分析を進めていく。

このトランスクリプトの中でも、発話のフィラーや笑い、引き伸ばしや発話間の重なりをどのように表現するかについて、様々な記法が開発されている。

本稿では、主に社会学の会話分析で利用される、Jefferson[4]に基づく(いわゆる Jefferson スタイルの)トランスクリプトの作成支援プログラムである“tracrin”の紹介を行う。Jefferson スタイルの特徴は、フィラーや笑いといった発話の種類分けを書き起こし時点では行わない点と、縦軸と横軸を用いて発話の重なりを視覚的理解しやすいうように提示できる点と考えられる。前者の特徴については小磯ら[5]による CSJ 方式の書き起こしと比較するとわかりやすいだろう。CSJ 方式では、フィラー発話は「(F_えーとー)」と記載するが Jefferson スタイルでは「え::と↑お」と記載する。前者は発話がフィラーであることが明確にしめされており、データベース内からフィラー発話を検索する際、利便性が高い。一方で後者は、フィラーか否かはわからないが、実際の発話をできるだけ表現しており、かつ(記法になれば)可読性が高い状態で表現している。

```
01 A k(.)こんばんわ::
02 B          [おつかれさま
03 C          [おつかれ
04 A          [そんな仕事してないよ
```

図1 Jefferson スタイルによるトランスクリプト作成例

後者の特徴を説明するため、実際のトランスクリプトを例示する(図1)。図1は左から行番号、発話者、発話内容が表記されている。発話内容の列では、発話をされた音声情報をできるだけ忠実にしめすため、笑いや呼吸を示す“(h)”, 音の伸長を示す“:”, 沈黙を示す“(n.m)”

といった記号が利用される。本稿では、その中でも発話の重なる開始を示すために用いられる“[”という記号と、重なる終わりを示すために用いられる“]”という記号について着目する。

この重なりを示す記号を用いることによって、単純な縦軸以外の時系列を横方向に示すことができる。図1では、01行目のAの「こんばんは」という発話内の「んわ」と02行目のBの「おつかれさま」という発話が重なって産出され、続けて「おつかれさま」の「さま」と重なってCが「おつかれ」と発話し、さらにCの「おつかれ」の「れ」と重なってAが「そんな仕事してないよ」と発話している、という構造が縦横の時間軸を用いて、発話を単位ごとに平面上に示されている。このように、発話の時間構造を直感的に理解しやすく、可読性の高いトランスクリプトが作成できることがジェファソンスタイルの特徴といえるだろう。

ジェファソンスタイルおよび、それを拡張したトランスクリプトを作成し提示するときに重要なのは、可読性が高く、かつ研究の目的にそった適切なトランスクリプトを読者に提示することである[6]。

一方で、同じ内容のトランスクリプトであっても、可読性が高いトランスクリプトを提示するためには、プレゼンテーションか論文なのかによって、また論文であっても掲載されるページのサイズの違いといった公開媒体によって、文字の大きさ、行数の文字数などを微妙に調整しなければならない。例として図1のトランスクリプトのフォントサイズや種類をずらしたものが図2となる。図2の右側のトランスクリプトは、“[”の位置が微妙にずれるため、適宜調整が必要な状態となってしまう。

01 A k(.)こんばんわ::		01 A k(.)こんばんわ::
02 B [おつかれ]さま	→	02 B [おつかれ]さま
03 C [おつかれ]		03 C [おつかれ]
04 A [そんな仕事してないよ]		04 A [そんな仕事してないよ]

図2 フォントの種類・サイズ変更によるズレ

このように、媒体に合わせて“[”の位置を調整することは、研究や分析にとって本質的ではない作業であり、tracrin は、上記の“[”の位置調整を自動化することを目的とした記法とプログラムである。

2. チュートリアル

本節では、tracrin の仕様や詳細な使い方ではなく、プログラミングに触れたことがないことを想定して、も

っともシンプルに使うための方法を紹介する。なお tracrin は Python3 で作成されたプログラムであり、2020/6/30 の時点で macOS BigSur 環境の Python3.8.2 の利用を確認している。macOS ではデフォルトで Python2 が入っているため、Python3 への切り替えが必要である。Python3 への切り替え方法は、様々な手段があるが、もっとも簡単な方法として、Anaconda の導入が存在する。

Anaconda は Python の統合開発環境の一つであり、をダウンロード・インストールが GUI ベースで実施可能である²⁾。以下、anaconda ベースの Python3 が導入されていることを前提に説明を続ける。

ベースとなるプログラムは tracrin_beta.py ファイルである³⁾。tracrin を利用するためには、python3 の環境構築(ここでは Anaconda の導入)、tracrin_beta.py と ELAN から出力した text ファイルが必要となる。tracrin を利用することで ELAN から出力した text ファイルから、“[”の位置を調整した CSV ファイルと text ファイルを生成することができる。

本節では、まず ELAN ファイルから出力された text ファイルを入力とし、発話内容のインデントが自動調整された CSV ファイルを出力するまでの手順として、ELAN への記法、ELAN から txt ファイルの出力、そして tracrin の利用の3手順について順を追って説明する。

2.1 ELAN の記法と出力

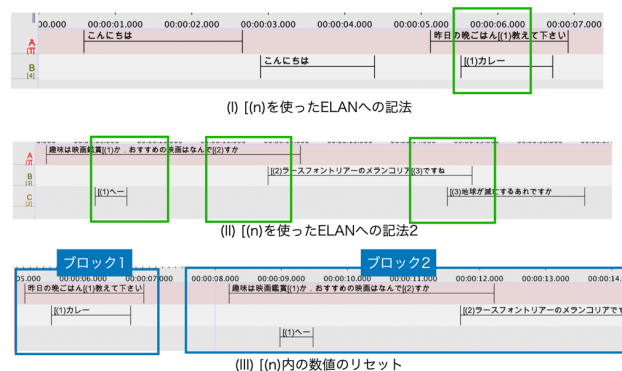


図3 ELAN 上での基本的な記法

まず tracrin を利用するために必要となる、ELAN 上での記法と出力設定について説明する。基本的に ELAN 上での発話の書き起こしは ELAN の基本機能を利用し[7]、発話内容を示すために使われる記号のほとんどは Jefferson スタイルに準ずる。本システムにおい

て、大きく異なるのは“[”記号の使い方のみである。発話の重なる開始を示すときには“[(n)”と記す(nには半角数値)を入れる(図 3-I)。n に入れる半角数値は任意であるが、同じ数値の入った箇所は重なった箇所を示す。例えば図 3-II)のように、3 つの箇所で発話の重なりがあったときには“[(1)”, “[(2)”と“[(3)”のように異なる数値を使って、それぞれの重なる位置を示すことができる。なお、例ではすべて (1)から開始していたが、“[(20)”, “[(32)”のように(1)以外から開始したり、連番でなくても問題はない。

発話の重なりが続く限り、異なる数値を使う必要がある(図 3-II)。しかし、発話間の重なりが終わり、次の発話の重なりが開始されたときには、同じ数値を使っても構わない。このような発話の重なりが続く状態をブロックと呼称する。例えば図 3-III では A の「昨日の晩ごはん教えて下さい」と B の「カレー」の2つの発話が重なっているが、それ以降の発話とは重なっていない。そのため、この2つの発話のみでブロックが形成されている。続く A の「趣味は映画鑑賞...」、C の「へー」、B の「ラースフォントリアーの...」の3つの発話は、前の「昨日の晩ごはん教えて下さい」と「カレー」の2つの発話とは時間的な重なりがまったくない。それ故図 3-III 内では、2つのブロックが形成されているとみなす。そして、それぞれのブロック内では、同じ数値を用いてはいけないが、異なるブロック間では、同じ数値を用いても構わない。つまり、ブロックごとに数値がリセットされているといえる。以上のようにして作成した ELAN ファイルを図 5 の設定で、タブ区切り文書形式で保存する。

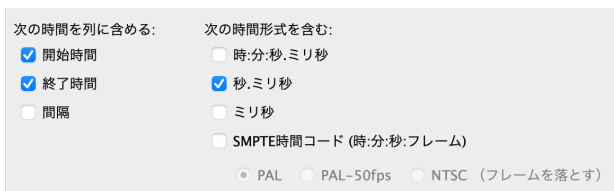


図 5 ELAN の出力設定

2.2 ターミナルによる tracin の操作

tracin_beta.py と上記の記法で用意した ELAN から出力したファイル(以下、dammy.txt とする)を同一ディレクトリに置いた状態を作る。ターミナルを操作し、そのディレクトリへ移動して、「python tracin_beta.py dammy.txt 04」と入力し実行する。実行後、同一ディレクトリ内に「dammy-notnum.csv」と「dammy-notnum.txt」

が生成される(図 5 と図 6)。「dammy-notnum.csv」では、右列から行数 (ID)、アノテーション種類(Label)、産出者(Speaker)、開始時間(Start)、終了時間(End)、発話内容(Content)となっている。発話内容箇所は前の行 “[” の位置までに文字数分の半角空白スペース⁴⁾が挿入されている。全角文字は半角空白スペース 2 つ分として計算して挿入されている。「dammy-notnum.txt」を word などに等幅フォントで貼り付けることで、“[” が適切な位置に調整されたファイルを作成することが可能となる。

ID	Label	Speaker	Start	End	Content
1	A_utter	A	0.59	2.66	昨日の晩ごはん[教えて下さい
2	B_utter	B	2.9	4.39[カレー.....
3	A_utter	A	5.12	6.92	趣味は映画鑑賞[か、おすすめの映画はなんで[すか
4	B_utter	B	5.525	6.725[へー.....
5	A_utter	A	8.21	12.22[ラースフォントリアーのメランコリア[ですな
6	C_utter	C	8.98	9.48[地球が滅亡する[あれですか
7	B_utter	B	11.71	14.935[見ると元気になるよな

図 6 ファイル 4 notnum の CSV ファイル⁴⁾

```

1 A      昨日の晩ごはん[教えて下さい
2 B      .....[カレー.....
3 A      趣味は映画鑑賞[か、おすすめの映画はなんで[すか
4 B      .....[へー.....
5 A      .....[ラースフォントリアーのメランコリア[ですな
6 C      .....[地球が滅亡する[あれですか
7 B      .....[見ると元気になるよな
    
```

図 7 ファイル 4 notnum の text ファイル⁴⁾

3. tracin 処理手順と応用的利用

本節では、tracin の処理手順を紹介すると共に、チュートリアル内では触れなかった応用的な利用と、想定した処理とはならない ELAN への記法と出力例について紹介する。

3.1 tracin における処理手順

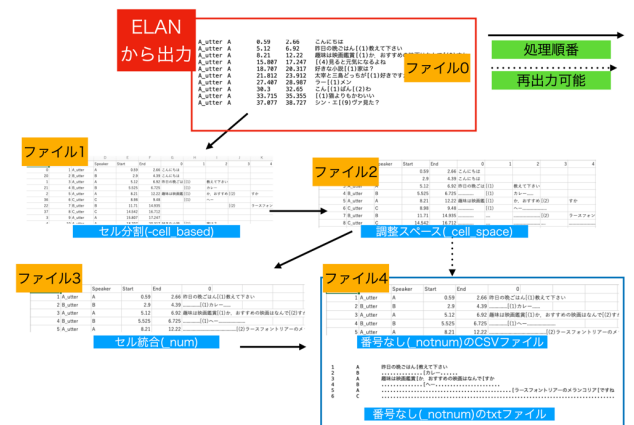


図 8 出力遷移とファイルの種類

tracin では、ELAN で出力したテキストファイルから 4 種のファイルの生成が可能である。これらのファイルは tracin の処理手順に対応している(図 8)。それぞれの

4.1 アップデート予定

今後のアップデートとしては(1)ファイル間の再出力に関するアップデート, (2) “[n]” 記号以外に対応させるアップデート, (3)1 行の文字数を指定して出力させるアップデートを予定している。

(1)現時点では, ELAN から生成されたファイル 0 からファイル 1, 2, 3, 4 への出力とファイル 2 からファイル 4 への出力のみを実装している。今後のアップデートで, ファイル 1 から 2, 3, 4, ファイル 2 から 3 への再出力を可能とする予定である。

(2)本システムでは, 発話内容に含まれる “[n]” 記号を用いて, 発話をセグメントとして区切り, 先行発話内で同一記号のある箇所にセグメントを調整する。このとき “[n]” 以外の記号を指定できるアップデートを予定している。近年では, Mondada[8]によって, 発話と身体的振る舞い間の時間的關係性を示すための記号が整理され, 多くの研究で利用されている。Mondada のシステムで利用される記号は “*”, “+” などである。これらについても “[” と同様に, “(n)” をつけることで, インデントの自動調整が可能になると考えられる。また, “{A}” のような任意の記号で発話と振る舞いの開始時点を示すこともある。そちらについても, 発話内容や既存の記号と重複しなければ, 検索記号に “[n]” 以外の任意の記号を指定することで対応可能である。

(3)現時点では, ELAN 上の 1 注釈に記載された文字数を 1 行の文字数としている。だが前述のように, 1 行の文字数は媒体によって変化しうる。現状ではファイル 2 を直接操作し, その後ファイル 4 を再出力することで 1 行の文字数が多すぎる箇所の調整が可能であるが, 今後 1 行の文字数を事前に設定し, 設定した文字数で行を分割した後に, インデント調整処理をする機能を実装する予定である。

4.2 今後追加実装予定のシステム

現状の tracin では, 記号に基づいたセグメント分割と, 分割後インデント調整により空白スペースの挿入を実施している。しかしながら, 全角 2 文字, 半角 1 文字として計算した半角空白スペース挿入によるインデント調整は, 必ずしも綺麗に調整されるわけではない。今後はファイル 1 のセグメント区切りの状態から, 発

表媒体に合わせた大きさの画像ファイルを出力することが望ましいと考えられる。

画像ファイルの作成方法としては, 様々なものが考えられる。一つの方法として Python の GUI パッケージの利用が挙げられるだろう。図 6 は Python の GUI の一つである Kivy を用いて tracin の結果を出力したものである。今後は, GUI を用いることで, 横幅などを任意に指定し, その幅に合わせた 1 行の文字数でのインデント調整を実施する可能性について検討していきたい。

10	A	好きな小説[家は？
11	C[武田泰[淳
12	B[ひかりごけの
13	A	太宰と三島どっちが[好きですか
14	B[どっちも友人にはしたくない

図 6 GUI による出力

さらに, GUI によりトランスクリプトを出力する場合は, 動画との連携が可能となる。例えば, 該当行をクリックすることで, 動画の該当箇所を再生する機能を実装可能である。会話分析においては, 多人数によるデータセッションが重要である。近年では, 遠隔でのデータセッションのやり方が模索されているが, 動画再生のタイムラグなどが問題となっている。GUI システムを実装することで, 新たな遠隔データセッションの形を提示することが可能になると考えている。

4.3 データベースへの利用可能性

最後に, tracin 内で生成されるファイル 1 (セグメントで区切られたファイル) が会話データベースで利用可能かについて議論する。Jefferson スタイルは, その特徴として可読性を重視しているものの, データベースでの検索には適さない。一方で, セグメントに区切られたファイル 1 には, セグメントごとに発話の種類 (たとえばフィラーや笑い) などのメタ情報を付与することができると考えられる。会話分析で用いられるトランスクリプトは, 研究発表のための資料というだけではなく, 論旨を突き進めるための資源と位置づけられるものである。そして, トランスクリプトを作成するためには, 膨大な時間とデータセッションなど他者との共同作業を必要とする[8]。よって, 会話分析のトランスクリプトは大変貴重な研究資源だといえよう。

現時点の `tracrin` 自体では、これまで `word` などで作成されたトランスクリプトをデータベース化することはできない。しかしながら、このようなシステムを構築することは、様々な記法により作成され、貴重な研究資源である書き起こし様々な方式のデータと結びつけ、新たな研究を進めるために必要となりうる会話データベース構築の一步となりうると考えられる。

脚注

- 1) 日本語に対応した記号については西阪[10]が整理したルールが多く用いられている。
- 2) `Anaconda` については、以下の URL よりダウンロードが可能である。 <https://www.anaconda.com/>
- 3) `tracrin_beta.py` については、以下の URL より配布している。 <http://rmakino.com/tracrin/>
- 4) 本稿ではスペースは説明の可視性のため “.” に置き換えられている。実際の `tracrin_beta.py` では空白(半角スペースが付与される)。
- 5) チュートリアルでは説明を簡略化・具体化するために同一ディレクトリ内のファイル指定のみを説明したが、相対パスを利用した指定も可能である。

謝辞

本研究はJSPS 科研費 17K18330 の助成を受けたものである。

文献

- [1] 林宅男(2008). “談話分析のアプローチ 理論と実践”, 研究社
- [2] Kendon, A. (1990). “Coconducting interaction: patterns of behavior in focused encounters”, Cambridge University Press
- [3] Sacks, H., Schegloff, E. A., Jefferson, G., 1978. A simplest systematics for the organization of turn taking for conversation. , Schenkein, J. (Ed.), “Studies in the Organization of Conversational Interaction”, Academic Press, New York, pp. 7-55.
- [4] Jefferson, G (2004). Glossary of transcript symbols with an introduction. Lerner, G. H. (Ed.), “Conversation analysis: Studies from the first generation”, pp.13-23, Philadelphia: John Benjamins Pub.
- [5] 小磯花絵・西川賢哉・間瀬洋子(2006).転記テキスト, “日本語話し言葉コーパスの構築法” 国立国語研究所報告 124 pp.23-132 国立国語研究所.
- [6] Mondada, L., (2007). Commentary: transcript variations and the indexicality of transcribing practices, “Discourse Studies”, Vol.9 No. 6, pp.809-821.
- [7] 細馬宏通・菊地浩平(編)(2019). “ELAN 入門”, ひつじ書房

- [8] 串田秀也・平本毅・林誠(2017). 会話分析入門, 勁草書房.
- [9] Mondada, L., (2009). Emergent focused interactions in public places: A systematic analysis of the multimodal achievement of a common interactional space, “Journal of Pragmatics” Vol41, pp.1977-1997.
- [10] 西阪仰(2008). “分散する身体”, 勁草書房.