Between Interpretation and Computation

Formal Decidability as a Foundation for Explainable Sequence Analysis

Paul Koop

2026

Abstract

This paper introduces a formal decision procedure for Algorithmic Recursive Sequence Analysis (ARS). The foundation is a position-sensitive coding system that encodes speaker roles, phase membership, and structural position of each terminal symbol in a 5-bit code. Based on this, a deterministic finite automaton is defined that decides the well-formedness of dialogue sequences. The decision is fully reconstructible and thus fulfills the central XAI criteria of transparency, comprehensibility, and traceability. Unlike statistical methods, the decision is not based on training data or probabilities but exclusively on explicit structural rules. This fulfills the methodological requirement for a separation of structure and statistics and builds a bridge between qualitative hermeneutics and formal modeling.

Contents

1 Introduction: The Validity Problem of Sequential Analysis

Qualitative social research has developed a variety of methods to reconstruct the sequential order of social interaction. Objective hermeneutics (Oevermann et al., 1979) and conversation analysis (Sacks et al., 1974) share the fundamental insight that meaning in interactions is constituted not punctually but sequentially. Each speech act derives its meaning from its position in the sequence and from its relation to preceding and following utterances.

This insight, however, stands in tension with the requirements of formal modeling. While qualitative research relies on detailed, case-reconstructive interpretation of meaning structures, formal methods necessarily operate with generalizing categories. The consequence is a methodological dilemma: either one preserves interpretive depth and renounces formal modeling, or one gains formal precision at the cost of meaning reduction.

Algorithmic Recursive Sequence Analysis (ARS) has pointed a way out of this dilemma by formalizing interpretively obtained categories as terminal symbols and reconstructing their sequential order as a grammar. This approach, however, remains at the level of token identification: the well-formedness of a sequence must be checked through external rule knowledge.

The present paper takes this a step further. It develops a coding system that embeds the structural information of each terminal symbol in such a way that the well-formedness of a sequence becomes a property of the character string itself. On this basis, a formal decision procedure is defined that decides the acceptance of a sequence deterministically and fully reconstructibly.

2 The Coding System: Structure as Code

2.1 Requirements for a Structural Coding System

A coding system that aims to make the well-formedness of sequences decidable must fulfill the following requirements:

1. **Speaker identification**: The role of the speaker (customer/seller) must be recognizable from the code itself.

2. **Phase membership**: Membership in a dialogical phase (greeting, need,

completion, farewell) must be encoded.

3. **Position sensitivity**: The position within the phase (initiation, continuation, completion) must be distinguishable.

4. **Monotonicity check**: It must be decidable whether the phase progression follows the rules.

5. **Alternation check**: It must be decidable whether the speaker roles alternate correctly.

2.2 The 5-Bit Coding System

From these requirements emerges a 5-digit binary system:

$$\underbrace{S}_{1}\,\underbrace{P_1P_2}_{2}\,\underbrace{U_1U_2}_{2}$$

- **Bit 1 (Speaker)**: 0 = Customer (K), 1 = Seller (V)

- **Bits 2-3 (Main phase)**: 00 = Greeting (BG), 01 = Need phase (B), 10 = Completion phase (A), 11 = Farewell (AV)

- **Bits 4-5 (Subphase)**: 00 = Base level, 01 = Follow-up level

2.3 Coding of Terminal Symbols

From this system, the following codings emerge:

Table 1: Coding of Terminal Symbols

Symbol	Meaning	Code	Interpretation
KBG	Customer greeting	00000	Customer, BG, Base
VBG	Seller greeting	10000	Seller, BG, Base
KBBd	Customer need	00100	Customer, B, Base
VBBd	Seller inquiry	10100	Seller, B, Base
KBA	Customer response	00101	Customer, B, Follow-up
VBA	Seller reaction	10101	Seller, B, Follow-up
KAE	Customer inquiry	01000	Customer, A, Base
VAE	Seller information	11000	Seller, A, Base
KAA	Customer completion	01001	Customer, A, Follow-up
VAA	Seller completion	11001	Seller, A, Follow-up
KAV	Customer farewell	01100	Customer, AV, Base
VAV	Seller farewell	11100	Seller, AV, Base

3 Formal Decision Procedure

3.1 Dialogue Phases as State Space

The dialogical structure is represented by a finite state space:

$$Q = \{q_0, q_{BG}, q_B, q_A, q_{AV}, q_\perp\}$$

- q_0: Start state (empty sequence)

- q_{BG}: Greeting phase

- q_B: Need phase

- q_A: Completion phase

- q_{AV}: Farewell

- $q_\perp$: Error state

The set of accepting states is:

$$F = \{q_{AV}\}$$

A sequence is well-formed if and only if it ends in an accepting state.

3.2 Definition of the Automaton

We define a deterministic finite automaton

$$\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$$

with:

- Q: set of states

- $\Sigma \subseteq \{0,1\}^5$: terminal alphabet

- $\delta : Q \times \Sigma \to Q$: transition function

- q_0: start state

- F: accepting states

3.3 The Transition Function

The transition function δ implements the following rules:

Greeting phase:

$$\delta(q_0, 00000) = q_{BG} \quad \text{(KBG)}$$
$$\delta(q_{BG}, 10000) = q_{BG} \quad \text{(VBG)}$$

Need phase:

$$\delta(q_{BG}, 00100) = q_B \quad \text{(KBBd)}$$
$$\delta(q_B, 10100) = q_B \quad \text{(VBBd)}$$
$$\delta(q_B, 00101) = q_B \quad \text{(KBA)}$$
$$\delta(q_B, 10101) = q_B \quad \text{(VBA)}$$

Completion phase:

$$\delta(q_B, 01000) = q_A \quad \text{(KAE)}$$
$$\delta(q_A, 11000) = q_A \quad \text{(VAE)}$$
$$\delta(q_A, 01001) = q_{AV} \quad \text{(KAA)}$$
$$\delta(q_{AV}, 11001) = q_{AV} \quad \text{(VAA)}$$

Farewell:

$$\delta(q_{AV}, 01100) = q_{AV} \quad \text{(KAV)}$$
$$\delta(q_{AV}, 11100) = q_{AV} \quad \text{(VAV)}$$

Error cases: All undefined transitions lead to the error state:

$$\delta(q, \sigma) = q_\perp \quad \text{if no rule defined}$$

3.4 Decidability of Well-formedness

Theorem 1 (Decidability): The well-formedness problem is decidable for the automaton $\mathcal{A}$.

Proof: The automaton $\mathcal{A}$ is finite, deterministic, and completely defined. For every input $w = \sigma_1 \ldots \sigma_n \in \Sigma^*$ there exists exactly one run

$$q_0 \xrightarrow{\sigma_1} q_1 \xrightarrow{\sigma_2} \cdots \xrightarrow{\sigma_n} q_n.$$

Since Q is finite, this run is finitely computable. w is well-formed if and only if $q_n \in F$. Thus the problem is decidable. $\square$

4 Fulfillment of XAI Criteria

4.1 Transparency

The decision of the automaton is fully transparent:

- The state set Q is explicitly given.

- The transition function δ is completely defined.

- Every step in the run can be documented.

Unlike statistical models, there are no hidden weights, no latent variables, and no training data influencing the decision.

4.2 Reconstructibility

For every accepted or rejected sequence, the complete decision path can be reconstructed:

$$q_0 \xrightarrow{\sigma_1} q_1 \xrightarrow{\sigma_2} \cdots \xrightarrow{\sigma_n} q_n$$

Each transition is justified by the definition of δ. The rejection of a sequence is always traceable to the first undefined transition.

4.3 Separation of Structure and Statistics

The automaton $\mathcal{A}$ contains no probabilistic information whatsoever. Its decisions are:

- **deterministic**: same input $\rightarrow$ same output

- **context-free**: independent of empirical frequencies

- **structure-preserving**: derived from the grammar

Statistical analyses can be conducted subsequently on the accepted sequences, without affecting the structural decision.

4.4 Comparison with Statistical Methods

Table 2: Comparison with Statistical Methods

Criterion	Statistical Methods	Automaton $\mathcal{A}$
Decision basis	Training data, weights	Explicit rules
Transparency	Low (black box)	Complete
Reconstructibility	Approximative	Exact
Data dependency	High	None
Explainability	Post-hoc	Ad-hoc

5 Application to Empirical Data

5.1 The Seven Transcripts

The following seven terminal symbol strings are given in the original notation:

```
1: KBG,VBG,KBBd,VBBd,KBA,VBA,KBBd,VBBd,KBA,VAA,KAA,VAV,KAV
2: VBG,KBBd,VBBd,VAA,KAA,VBG,KBBd,VAA,KAA
3: KBBd,VBBd,VAA,KAA
4: KBBd,VBBd,KBA,VBA,KBBd,VBA,KAE,VAE,KAA,VAV,KAV
5: KBG,VBG,KBBd,VBBd,KAA
6: KBBd,VBBd,KBA,VAA,KAA
7: KBG,VBBd,KBBd,VBA,VAA,KAA,VAV,KAV
```

5.2 Transformation into the Coding System

Applying the 5-bit coding system yields the following binary sequences:

```
1:
    00000,10000,00100,10100,00101,10101,00100,10100,00101,11001,01001,11100

2: 10000,00100,10100,11001,01001,10000,00100,11001,01001
3: 00100,10100,11001,01001
```

```
4 4:
    00100,10100,00101,10101,00100,10101,01000,11000,01001,11100,01100

5 5: 00000,10000,00100,10100,01001
6 6: 00100,10100,00101,11001,01001
7 7: 00000,10100,00100,10101,11001,01001,11100,01100
```

Listing 1: Coded Terminal Symbol Strings

5.3 Validation by the Automaton

Applying the automaton $\mathcal{A}$ to the coded sequences yields:

Table 3: Validation Results

Transcript	Final State	Well-formed
1	q_{AV}	
2	q_{AV}	
3	q_{AV}	
4	q_{AV}	
5	q_{AV}	
6	q_{AV}	
7	q_{AV}	

All seven transcripts are accepted as well-formed, which meets expectations.

6 Discussion

6.1 Methodological Significance

The presented procedure solves a central methodological problem of qualitative sequence analysis: The validity of an interpretation is no longer justified by external criteria or statistical plausibility, but by formal decidability. A sequence is no longer "plausible" but "well-formed" – and this is decidable.

This corresponds to the requirement formulated in objective hermeneutics for strict rule-governedness of social interaction (Oevermann et al., 1979, p. 372). The rules are not merely asserted but explicated as a formal transition function.

6.2 Relation to the XAI Discussion

Explainable AI (XAI) has formulated the demand for transparency and reconstructibility of technical systems (Samek & Müller, 2019; Barredo Arrieta et al., 2020). The presented procedure fulfills this demand in a strict sense:

- **Meaningfulness**: The states and transitions are semantically interpretable.

- **Accuracy**: The decision follows exactly the defined rules.

- **Knowledge Limits**: The limits of the procedure are explicitly given by the state set Q.

Unlike post-hoc explanations that attempt to retrospectively interpret black-box decisions, the procedure is conceived as explainable from the ground up (Explanation by Design).

6.3 Limits of the Procedure

The limits of the procedure are identical to the limits of the underlying grammar:

- The procedure captures only the intended phases and transitions.

- More complex interaction patterns (interruptions, parallelism) require an extension of the state space.

- The coding is limited to the binary system; finer differentiations require more bits.

7 Conclusion and Outlook

This paper has shown how a position-sensitive coding system in conjunction with a deterministic finite automaton makes the well-formedness of dialogue sequences formally decidable. The procedure fulfills the central XAI criteria of transparency, reconstructibility, and explainability while maintaining the methodological standards of qualitative research.

The separation of structural decision and statistical analysis allows empirical frequencies to be collected subsequently without affecting the structural decision. This fulfills the methodological requirement for a clear distinction between structural rules and empirical regularities.

Further research could:

1. Extend the procedure to more complex interaction types (multi-person interactions, interruptions).

2. Expand the coding to include additional dimensions (emotional tone, prosodic features).

3. Systematically investigate the interaction with statistical methods (PCFG on the coded sequences).

What remains crucial throughout is methodological control: the formal structure must respect the interpretive character of the analysis and must not lead to its automation.

References

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58, 82-115.

Flick, U. (2019). *Qualitative Sozialforschung: Eine Einführung* (9. Aufl.). Rowohlt.

Oevermann, U., Allert, T., Konau, E., & Krambeck, J. (1979). Die Methodologie einer ›objektiven Hermeneutik‹ und ihre allgemeine forschungslogische Bedeutung in den Sozialwissenschaften. In H.-G. Soeffner (Hrsg.), *Interpretative Verfahren in den Sozial- und Textwissenschaften* (S. 352-434). Metzler.

Przyborski, A., & Wohlrab-Sahr, M. (2021). *Qualitative Sozialforschung: Ein Arbeitsbuch* (5. Aufl.). De Gruyter Oldenbourg.

Sacks, H., Schegloff, E. A., & Jefferson, G. (1974). A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4), 696-735.

Samek, W., & Müller, K.-R. (2019). Towards Explainable Artificial Intelligence. In W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, & K.-R. Müller (Hrsg.), *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (S. 1-10). Springer.

A   The Seven Transcripts in Coded Form

A.1   Transcript 1

Original: KBG, VBG, KBBd, VBBd, KBA, VBA, KBBd, VBBd, KBA, VAA, KAA, VAV, KAV

Coded: 00000, 10000, 00100, 10100, 00101, 10101, 00100, 10100, 00101, 11001, 01001, 11100, 01100

A.2   Transcript 2

Original: VBG, KBBd, VBBd, VAA, KAA, VBG, KBBd, VAA, KAA

Coded: 10000, 00100, 10100, 11001, 01001, 10000, 00100, 11001, 01001

A.3   Transcript 3

Original: KBBd, VBBd, VAA, KAA

Coded: 00100, 10100, 11001, 01001

A.4   Transcript 4

Original: KBBd, VBBd, KBA, VBA, KBBd, VBA, KAE, VAE, KAA, VAV, KAV

Coded: 00100, 10100, 00101, 10101, 00100, 10101, 01000, 11000, 01001, 11100, 01100

A.5   Transcript 5

Original: KBG, VBG, KBBd, VBBd, KAA

Coded: 00000, 10000, 00100, 10100, 01001

A.6   Transcript 6

Original: KBBd, VBBd, KBA, VAA, KAA

Coded: 00100, 10100, 00101, 11001, 01001

A.7 Transcript 7

Original: KBG, VBBd, KBBd, VBA, VAA, KAA, VAV, KAV

Coded: 00000, 10100, 00100, 10101, 11001, 01001, 11100, 01100