

Introduction

Traditional methods for monitoring implementation of evidence-based programs require labor-intensive quality assessments. These assessments generally involve human observation, hence becoming a major bottleneck in the monitoring of implementation. We present the development of computer-based methods for implementation quality measurement.

Firstly, we describe 7 principles of automatization necessary to systematically identify critical dimensions for analysis.

Secondly, we present an automatic classifier that uses linguistic patterns to classify sessions between high-fidelity versus low-quality. The classifier was trained on an initial corpus of 22 transcripts which were coded for high and low quality by humans and replicated by machine learning. Our goal is to develop efficient methods that reduce the burden of quality monitoring, and provide timely feedback for delivery supervisors.

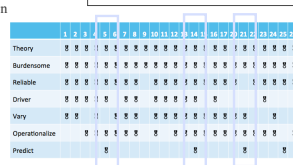
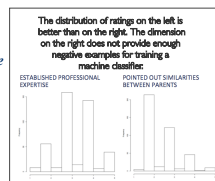
Overall, we want to explore to what extent quality monitoring can be done with automatic methods [1.3.4].

Methods

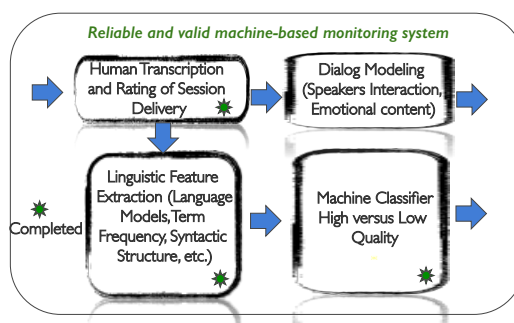
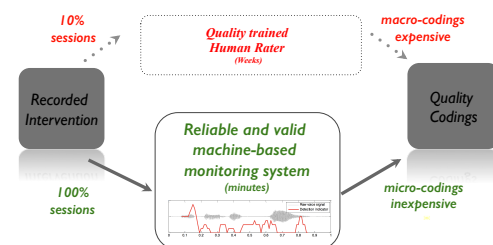
Based on human-based ratings of a subset of 22 sessions of the New Beginning Program (NBP) for divorcing parents [2], we identified linguistic behavior that the group leader delivering the program should follow: a) Provide helpful examples, b) Indicate belief in parent's ability. These dimensions met 7 **Automatization Principles**, including having the highest inter-rater reliability and having predicted the relevant outcomes such as parent's retention and positive engagement in home practice. We extracted linguistic features from this set of transcripts and used a **Machine Learning KNN algorithm** to classify utterances where quality indicators demonstrated high and low quality for the targeted dimensions.

Automatization Principles: *Criteria for the identification of dimensions for machine monitoring.*

1. Be relevant to the program's theory
2. Be burdensome for human coding
3. Be reliable across raters ($>IRR .75$)
4. Be a driver of the dimension
5. Vary across the sample
6. Be operationalizable
7. Predict program outcomes



Machine-Based Method for Measuring Quality



Data

Creation of Corpus: Sessions of New Beginnings Program delivery was recorded, transcribed, and rated for quality by humans. From the transcripts, we selected the speech from the group leader, separated into utterances, and applied the session-level quality rating to each utterance. A sample of the data is shown in the table below. In this example, there are two sessions transcripts, six utterances from group leader, and six quality ratings (3 high and 3 low).

Session #	Utterance Number	Utterance	Quality Human-Bas Dimension #21 Indicated Belief
S1	Ut1	that was a pretty good solution you came up with. So, congratulations on sticking through with it	High
S1	Ut2	It's really cool to hear that. I mean at least everyone was giving it a shot, and to me it sounded like overall it's going positive.	High
S1	Ut3	You want to be there and you want to trust yourself, 'cause you got it.	High
S2	Ut1	It was all because of her	Low
S2	Ut2	but I didn't really get to an open question. My mind just went blank	Low
S2	Ut3	Maybe that'll help you use the skills	Low

Machine Learning Classifier

Training and Testing a machine classifier. There are a series of steps necessary to train and test a machine classifier. In general terms, we need to:

1. Create the Corpus
2. Preprocess of the Corpus
 - Stem words
3. Extract features for KNN Model [5]
 - Create a Term Document Matrix
4. Selecting the training & testing sets
5. Train the model
6. Test the model

High Quality Utts (Corpus A)

	Docs														
Terms	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
and	7	0	1	1	0	0	1	1	2	0	0	0	0	1	0
can	0	1	1	1	0	0	0	0	0	0	0	1	0	1	0
have	1	0	1	2	0	1	0	1	3	0	0	0	1	1	0
the	5	0	2	5	1	1	0	1	0	0	0	3	0	0	0
this	12	0	1	0	0	1	0	2	2	0	0	0	2	0	0
what	1	0	0	1	1	0	0	0	0	0	0	0	0	1	1
when	1	0	0	0	0	1	1	0	0	0	1	0	1	0	0
you	1	1	1	0	0	0	0	0	2	2	0	0	0	0	0

Novel Transcript

KNN
Machine Classifier

- For any given new document
- The classifier extracts
- Linguistic features
- And compares them with the
- Training examples (A or B).
- KNN: K nearest neighbors
- Algorithm chooses closest
- Category in the
- Feature space

Identified Category
(High/Low)

Results and Discussion

Can machine learning recognize high versus low quality dimensions in session delivery transcripts?

Indicated Belief Dimension We found that based on 22 transcripts that were coded for Indicated Belief quality dimension, the KNN (K-Nearest Neighbor) algorithm **correctly classified 83%** of the utterances in 10 testing documents. This results suggests that monitoring quality can be effectively with computational methods.

Future Work

We will apply these methods to other dimensions such as “Provide Helpful Examples.” Our goal is to be able to rate quality directly from the audio signal without the need of human transcription. First, we demonstrate that linguistic features can be measured automatically. Second, we need to adapt, test, and validate automatic transcription for measuring quality. Another area to explore is the interaction between group leaders and parents. Dialog Modeling can provide other measurable metrics that correlate with quality of delivery.

References

- [1] Brown CH, Mohr DG, Gallo GC, Mader C, Sakka LA, et al. A computational theory for preventing HIV in minority communities: How advanced technology can improve implementation of effective programs. *JAIDS Journal of Acquired Immunity Deficiency Syndromes*. 2013.
- [2] Berkei C, Mauricio A, M., Schoenfelder E., & Sandler, I. N. (2011). Putting the pieces together: An integrated model of program implementation. *Prevention Science*, 12(1), 23–33.
- [3] Gallo GC, Pappa M, Vittinghoff E, Grimsley J, Probst C, et al. (2012). Linguistic and Computational Linguistics Methods for Fidelity Assessment: Experience with the Family AIDS Prevention Intervention. *Administration and Policy in Mental Health*. 2014.
- [4] Inoue, M., Ogihara, M., Hanada, R., & Furuyama, N. (2012). Gestural cue analysis in automatic semantic miscommunication annotation. *Multimedia tools and applications*, 61(1), 1–20.
- [5] Lagan, D. (2012). "k-Nearest Neighbor Algorithm." *Discovering Knowledge in Data: An Introduction to Data Mining* (2005): 90–106.