A Case Study: Detecting Counselor Reflections in Psychotherapy for Addictions using Linguistic Features

Doğan Can[1]*, Panayiotis G. Georgiou*[2]*, David C. Atkins*[3] *and Shrikanth S. Narayanan*[1,2]

[1]Department of Computer Science and [2]Department of Electrical Engineering,
University of Southern California, Los Angeles, CA, USA
[3]Department of Psychiatry and Behavioral Sciences, University of Washington, Seattle, WA, USA
dogancan@usc.edu, georgiou@sipi.usc.edu, datkins@u.washington.edu, shri@sipi.usc.edu

Abstract

Motivational Interviewing (MI) is a goal-oriented psychotherapy, employed in cases such as addiction, which helps clients (i.e., patients) explore and resolve their ambivalence about the problem at hand in a dialog setting. Measuring the counselor's proficiency with MI has typically been assessed via behavioral coding - a time consuming, non-technological approach. This paper examines a computational approach to assessing the quality of MI. Specifically, we focus on a particular aspect of the counselor behavior – reflections – believed to be a critical indicator of MI therapy quality. We automatically tag reflection instances in a maximum entropy Markov modeling framework using several linguistic features with rich contextual information obtained from the session transcripts. We achieve an F-score of over 80% while gaining insight about the information sources as perceived by the trained annotators.

Index Terms: dialog act tagging, behavioral signal processing, motivational interviewing skills code

1. Introduction

Behavioral Signal Processing (BSP) is an emerging area that aims to model, quantify and give insights into complex human behaviors. It combines computational techniques from multimodal signal processing, natural language processing and machine learning with psychology and behavioral sciences expertise to offer computational tools and insights beyond what is conceptualized in theory about the human behavior, its expression and its perception. The mental health domain, and in particular observational studies of human behavior, constitutes one of the primary application domains targeted by BSP.

Motivational Interviewing (MI) is a non-confrontational, directive counseling approach extensively used in treating alcohol and other drug-related problems [1, 2]. Many individuals struggling with addictions perceive both benefit (e.g., the "high") and harm (e.g., missing work, problems with spouses) from their use and are ambivalent about changing their behavior. MI focuses on eliciting and enhancing the intrinsic motivation for change by exploring and resolving client ambivalence in a dyadic spoken dialog setting. MI is client-focused; it has an empathic focus and emphasizes the client's right and responsibility to make changes related to their addictive behaviors. There is a strong evidence-base for MI [3], and a current focus in mental health policy research is how to support the effective implementation and dissemination of MI into community treatment centers. One significant challenge is how to ensure high quality treatment. Heretofore, the typical method for assessing quality, or proficiency, of MI is to use a behavioral coding system in which human raters learn and then utilize a system for annotating video or audio tapes. However, this approach does not scale up to real-world use [4]. Thus, an automated solution to coding – and hence assessment of MI quality – would greatly enhance the dissemination of high-quality MI into the community.

The Motivational Interviewing Skills Code (MISC) is an observational coding method for evaluating the quality of MI from audio- and video-tapes of individual counseling sessions [2]. It is designed for manually annotating an interview between two individuals, the Counselor and the Client. This annotation is performed by trained coders who associate an appropriate behavioral code to each counselor and client utterance in addition to assigning session-level global ratings to characterize the entire interaction. Utterance-level codes are similar in function to dialog acts that are often used to annotate high-level discourse structure of a dialog. We view the utterance-level MISC coding process as a specialized form of the standard NLP task of dialog act tagging and in the longer-run aim to develop an automatic system that will assist MI researchers and practitioners in this phenomenally labor intensive process. As a first step, in this work we turn our attention to detecting a particular class of counselor responses termed *reflections*.

Reflections constitute a major class in the MISC hierarchy and they are particularly interesting to study since they encode a non-trivial counselor behavior, known as reflective listening, which is believed to be one of the fundamental elements of MI efficacy. The MISC manual [2] defines a reflection as "a reflective listening statement made by the counselor in response to a previous client statement." Their function is "to capture and return to the client something that the client has said." They can "simply repeat or rephrase what the client has said," introduce "new meaning or material," or "summarize part or all of a session." Reflections tend to be collaborative and non-judgmental and aim to guide the client towards resolving ambivalence.

From an NLP perspective, a reflection pair, reflection along with the client speech that is being reflected, is quite similar to a textual entailment pair [5] in the sense that the reflection is entailed by what the client has said before. The primary difference is that in the case of a reflection pair, the entailment chain may require incorporating expert knowledge that is not present in the dialogue. However, unlike the standard textual entailment task, which uses a given set of entailment pairs for learning and another set of tentative entailment pairs for evaluation, a semantic approach to reflection detection requires figuring out the associated client speech in the dialogue both during learning and evaluation. On the flip side, reflections are not as linguistically diverse as the standard textual entailment statements. Counselors tend to use certain linguistic constructs while reflecting which we leverage in this work to elicit reflections from other counselor speech. It should be noted that this is a non-trivial task which requires dealing with the variability in reflection generation. Another important aspect is that reflections occur in dialogue, which means they not only affect the dialog flow, e.g. clients tend to confirm a reflection with statements like "yeah", but also are affected by what is happening in the local context, e.g. reflections tend to occur immediately after a client utterance as opposed to a counselor utterance, reflective utterances tend to occur in succession, etc. From this perspective, a reflection is like a dialog act and can potentially be inferred by examining the local context. In this paper, we limit our attention to local context and present a maximum entropy Markov model based tagging system which uses automatically extracted linguistic features with rich contextual information to detect reflection occurrences.

Since reflections are the focus of this study, we structure our analysis around their definition and investigate the following sources of information: the speaking style of the counselor ("summarize", "listen reflectively", "be collaborative and non-judgmental") [Sec. 3.1], the response of the client ("confirmation") [Sec. 3.1], the content of the counselor response in relation to prior client talk ("capture and return") [Sec. 3.2], and the dialog flow ("in response to previous client statement") [Sec. 3.3].

2. Background and Data

2.1. Motivational Interviewing Skills Code (MISC)

The MISC coding procedure consists of annotations at two different resolutions: session-level global ratings on a 7-point Likert scale and utterance-level behavioral codes. Each utterance is assigned one of the 20 distinct behavior codes (15 counselor + 5 client) given in Table 1. These codes are intended to capture specific local behaviors within the dialogue. While the global context might influence the coder decisions, in general the codes are determined by considering the local context.

The MISC defines an utterance as a *complete thought* and it ends either when the speaker completes a thought and moves to another or when the speaker changes. Utterance boundaries are decided by coders in the process of coding, and thus, there is some disagreement across coders in defining utterance boundaries. The beginning and end of talk-turns are hard utterance boundaries, but multi-utterance talk-turns can lead to differing numbers of utterances – hence codes – for each coder. This fact complicates measuring inter-coder reliability which we will say more about in the next section.

Table 1: MISC Categories and Counts

AD	Advise	331	RE	Reflect	7115
AF	Affirm	1384	RF	Reframe	13
CO	Confront	22	SU	Support	352
DI	Direct	6	ST	Structure	1004
EC	Emphasize Control	48	WA	Warn	6
FA	Facilitate	8839	FN	Follow/Neutral	29376
FI	Filler	29	R	Reason	2862
GI	Giving Information	9544	O	Other	2028
QU	Question	5220	TS	Taking Steps	86
RC	Raise Concern	9	C	Commitment	36

2.2. Reflections

Reflections are believed to be one of the most critical components of MI. They tend to have an empathic tone and can serve to reflect back to the client both positive and negative outcomes of their addictions – thus, helping the client to resolve (or at least face) their ambivalence. The MISC system classifies two types of reflections. Simple Reflections (RES) are those which add little or no meaning to what the client has said. Their primary function is to convey understanding. Repeating or rephrasing what the client has said is considered RES. Complex Reflections (REC) typically add substantial meaning or emphasis to what the client has said. They convey a deeper or richer picture of the client's statement and may contain significantly more or different content from what the client has actually said. The differences may be subtle or obvious. Analogies, metaphors, similes, exaggerations and summaries almost always fall under the REC category. Figure 1 gives several reflection examples that we borrow from the MISC manual [2]. Our goal in this paper is to automatically detect the reflection occurrences in dialogue. Our premise is that i) a common language or behavioral pattern is shared across reflections and ii) the local dialog context is both influenced by and at the same time influences a reflection occurrence. Our hypothesis is that reflections can be treated as dialog acts and can be inferred by detecting this common language use and examining what is happening in the local context.

Cli: I wouldn't mind coming here for treatment but I don't want to go to one of those places where everyone sits around crying and complaining all day.
Cou: You don't want to do that. **RES**
Cou: So you're kind of wondering what it would be like here. **REC**

Cli: At one time I was pretty much anti anything but marijuana.
Cou: Marijuana was OK. **RES**
Cou: That's where you drew the line. **REC**

Cli: Everyone's getting on me about my drinking.
Cou: Kind of like a bunch of crows pecking at you. **REC**

Figure 1: Reflection Examples

2.3. Data

Our MISC coded data-set includes sessions from three MI intervention studies focusing on drug abuse problems (HMCBI) and alcohol use disorders (ESP21, ESPSB). Prior to coding, each session audio-tape was carefully transcribed (marking backchannels, disfluencies, interruptions, overlaps, etc.) and annotated with turn-level time alignments. In its current status, our data set includes a total of 57 sessions. Some sessions were coded by multiple coders mainly to establish coder reliability and as a result we have obtained 108 coded sessions. Since utterance segmentation is performed by the coders, each coded session is unique not only in terms of the assigned codes but also the particular segmentation of turns into utterances.

In the previous section we alluded to the difficulties involved in calculating agreement statistics due to the fuzzy mapping between utterances. Since there is not an obvious way of comparing codes assigned by different coders, we calculated two different group "multi-κ" (average pairwise agreement) values that approximate the code reliability. For this purpose we used 12 sessions (10 HMCBI + 2 ESP21) independently coded by all three coders at the beginning of our study. The first approximation simply ignores the utterances that do not match in endpoints and uses utterance as the basic unit of comparison. The second approximation assigns each word the code associated with the parent utterance and uses word as the basic unit of comparison. The first approach resulted in a multi-κ agreement value of .764 which is on the moderate-to-high side. The second resulted in a value of .661 which is not surprising since using words as the basic units significantly increases the resolution and suffers from insignificant boundary mismatches. The actual code agreement value lies somewhere in between these two extremes and is well within the range of agreement values typically observed in MISC studies [6].

We performed standard normalization operations on our data set before we used them in the experiments of the next section. We removed all word-external punctation, i.e. we kept hyphens, apostrophes and underscores (used for keeping entities together), tokenized utterances in the Penn Tree Bank style, i.e. split on word boundaries, word-external punctuation and apostrophes, and finally lowercased everything. Table 2 summarizes various statistics of our experiment set.

Table 2: Experiment Data Statistics (Counts). Columns respectively represent the total number of speaker turns, unique (utterance, code) pairs, tokens and unique tokens.

Data	Turns	Tuples	Tokens	Types
All Coders	28723	45598	732052	8161
One Coder	15286	35187	428934	8161

3. Linguistic Features

In our tagging framework, we process utterances in sequence and extract features for each utterance considering a parametrized local context. Since reflections are the focus of this study, we will design our contextual features based on the reflection definition given in the previous section.

3.1. N-gram Features

Therapists tend to use certain linguistic constructs while reflecting, e.g. phrases like "from what I gather" and "it sounds like" are common across reflections. In the case of complex reflections, this becomes even more pronounced since almost all "analogies, metaphors, similes, exaggerations, summaries, etc." fall under the REC category. Furthermore, some other types of counselor and client categories tend to have their own specialized language patterns, e.g. question, giving information, reason. We use word n-gram features (for n $\leq N$) extracted from the current utterance to capture the particular language use.

A typical scenario is when the client responds to a reflection in a confirmatory manner, e.g. many client utterances following a reflection statement start with the discourse marker "yeah". This behavior is not distinguished by MISC and is usually coded as a *Follow/Neutral* (FN) statement. We use word n-gram features (for n $\leq N$) extracted from contextual utterances to capture this kind of phenomena.

Each n-gram feature is prefixed with the associated role ID (counselor or client) and the relative position of its host utterance with respect to the current utterance (before or after) so that two identical n-grams associated with different roles or different contextual positions are treated separately.

3.2. Similarity Features

Reflections' primary function is to "capture and return" the client talk. Especially in the case of simple reflections (RES), counselors tend to simply repeat or mildly rephrase what the client has said before. With that in mind, we gather all word-stem n-grams (so that morphological variation is normalized) for n $\leq N$ shared between a counselor utterance and the preceding client utterances in the local context (previous N_{C} client utterances), and then eliminate those which do not include at least one content word, i.e. a noun, verb, adjective or adverb. We use the remaining word-stem n-grams and the part-of-speech n-grams corresponding to them as binary similarity features.

3.3. Contextual Meta-Features

As with the standard dialog act tagging task, contextual role IDs and tags (in our case codes) can be helpful in discriminating reflections from other counselor speech. To that end, we extract role ID and code n-grams from the local context (N_{code} previous codes, N_{role} previous/next role IDs) and use them as another source of information. At the training time, we use the actual (reference) codes associated with each contextual utterance while during evaluation we replace those with the codes hypothesized by our tagger.

4. Experiments

Since we are working with a relatively small data set, on average 1760 utterances per code, we wanted to maximize the size of the training set used in each experiment. Hence, we adopted a leave-one-out cross validation approach and run the same experiment 57 times leaving one session out in each run. All results that we report in this section were obtained by taking the weighted average of individual runs where the weight is simply the number of counselor utterances in the left-out session. If a session was coded by multiple coders, we treated all coded variations of each session as a separate sample both during training and testing. To counter the bias introduced by including some sessions multiple times inside the training set, we created replica sessions for each coder that did not code a particular session and included them in our data set.

4.1. Baseline: Hidden Markov Model (HMM)

Our baseline performance figures come from an HMM tagger [7] in which latent states represent code categories in context. We experimented with mono- and tri-code states – akin to mono- and tri-phones in automatic speech recognition –, and trained a n-gram language model (for n $= N_{\mathrm{chnl}}$) for each state (of size N_{C}) by collecting all utterances in our training set that match the particular context. We interpolated mono-code models with a background language model and tri-code models with the smoothed mono-code models.

We combine the scores coming from the channel model (collection of n-gram models trained for each state) with the scores coming from the source model (an n-gram model trained over code sequences for n $= N_{\mathrm{src}}$) using a Viterbi decoder and search for the best code sequence that explains our observation (a complete session transcription) given the models.

4.2. Maximum Entropy Markov Model (MEMM)

MEMMs [8] are used extensively in sequence tagging tasks due to the relative ease of incorporating arbitrary features into these models. Here we adopt a MEMM framework and experiment with various combinations of contextual features parametrized by factors like the context sizes used for extracting contextual features, which can be different for each type of contextual feature, and the maximum n-gram sizes which again can be different for the current and the contextual utterances. Similar to what we did in the case of HMM, we combine the posterior scores assigned by the maximum entropy models using a Viterbi decoder and search for the best code sequence.

4.3. Feature Selection

Using all word n-gram features extracted from the local context creates a massive feature space in which other types of features constitute a tiny fraction. Our data-set is rather small to learn that many parameters. Hence, we experiment with limiting the word n-gram features used in our model to the most informative ones as determined by the relative entropy measure. We select the n-gram features that satisfy the following conditions: i) the relative entropy of the empirical distribution of that n-gram among the codes should be at most $E_{\max}$ and ii) it should occur at least $C_{\min}$ times in our training data set.

4.4. Code-Set Reduction

Not all code categories are equally represented in our data-set (see Table 1). To deal with the sparsity problems caused by this uneven distribution and the limited size of our data-set, we cluster all client codes into the meta-code CLI and all counselor codes except reflections into COU and keep RE as a separate code.

4.5. Results

Table 3 summarizes the performance of MEMMs under various feature parameterizations and combinations. Except for the last experiment, which incorporates code labels from the previous utterances, the MEMM reduces to a regular maximum entropy model, i.e. each utterance is tagged independently. The models given in Table 3 are the best performing ones for each feature combination we experimented with. Albeit their simplicity, similarity features are able to capture some of the reflection occurrences at an acceptable precision level of 67%; yet

Table 3: Feature Comparison (MEMM)

Feature[Parameters]	Rec. (%)	Prec. (%)	F-score (%)
Sim[4, 9]	31.14	67.05	42.53
N-gram[4, 0, 0.35, 2]	63.63	86.73	73.40
N-gram[4, 1, 0.35, 2]	94.14	69.86	80.20
Sim[4, 9] + N-gram[4, 1, 0.35, 2]	91.46	69.56	79.02
Meta[2, 2] + N-gram[4, 1, 0.35, 2]	92.53	72.69	81.42

Sim[N, N_{C}], N-gram[N, N_{C}, $E_{\max}$, $C_{\min}$], Meta[N_{code},N_{role}]

the recall is quite poor at 31%. While this result might be due to our simple approach to modeling utterance similarity, it appears like reflections do not repeat or mildly rephrase the client statements as often as suggested by the MISC guideline. Furthermore, the precision value suggests that other counselor talk (besides reflections) also shares content words with prior client speech, which should not come as a surprise since we are talking about a dialogue after all. Even without the Markovian setting, n-grams of the current utterance are able to represent the language use specific to reflections with an increased F-score of 73% (second row). This is even more pronounced when we include the n-grams coming from the immediate context at an F-score of 80% (compare second and third rows) which indicates that there is rich information buried in the local neighborhood. One aspect of this is the client response to a reflection. There are likely other contextual indicators such as the tone of the previous client utterance (opportunity for a reflection or for expressing empathy by the counselor) or the language used in a neighboring counselor utterance, e.g. successive reflections. We did not receive any gain from combining the similarity and the n-gram features. Adding meta-features, on the other hand, further improved the precision to 72% and F-score to above 81%. This gain is largely due to the the previous code n-grams which contextualize the current utterance within the dialogue.

Table 4 compares the best HMM results with the best MEMM results. Again these are the best HMM results we obtained. Note that contextual n-grams, which are not available to HMMs, constitute the primary difference between the information sources used by these models. Meta-features in MEMM play a similar role to the source language model in HMM.

5. Discussion

We presented a method for detecting reflections in psychotherapy sessions conforming to the Motivational Interviewing counseling approach. On the algorithmic side our best HMM results were below our best MEMM results, likely due to the contextual n-grams available to the latter method.

On the feature side we have compared several sources of information: similarity features that attempt to compare the lexical content of the counselor reflection with prior client talk; n-gram features that attempt to capture the speaking style; and contextual meta-features that attempt to capture the dialog flow. We observed that according to our classification schemes there is significantly higher information about the presence of reflections in the n-gram features, hence in the speaking style, rather than in the similarity or context of the dialog. This can point to one or several of the following reasons: sparsity issues, inability of statistical models and features to capture the information contained in signal, or it may denote the human process of information extraction, i.e. how something was said was more significant towards the perception of a reflection than the content. The latter of the three reasons is receiving careful consideration by the MI experts. Specifically we are interested in critically asking and addressing questions about the process of reflection, both its production (Is the style equally important as the content of the reflection? Is reflection a more local process?) and perception (Are coders affected by the saliency of specific speaking styles in their coding process?). In this part we hope that our classification scheme will go beyond aiding in the annotation process and provide feedback to the MI experts.

Table 4: Model Comparison

Model[Parameters]	Rec. (%)	Prec. (%)	F-score (%)
HMM[mono, 2, 5]	66.55	75.42	70.71
HMM[tri, 2, 4]	69.43	73.68	71.49
MEMM[2, 2, 4, 1, 0.35, 6]	92.53	72.69	81.42

HMM[N_C, N_{chnl}, N_{src}], MEMM[N_{code}, N_{role}, N, N_C, $E_{\max}$, $C_{\min}$]

6. Conclusions and Future Work

Our aim is two-fold: to aid the experts in their observations and to transform the observational methods. In this work we show early work towards extracting information about reflections from the lexical channel using manual transcripts. Past work [9] has established that this type of lexical information can also be extracted from ASR lattices. We plan to investigate this direction further in our future efforts. Towards transforming the observational practice, we are presenting here a first effort in understanding the perception process of reflections, critically raising questions for future work.

There are significant challenges and opportunities ahead, from the engineering challenges of extracting lexical features from the acoustic channel, to using and fusing these with acoustic features, to employing automatic speaker diarization and to the MI challenges of further understanding the production and perception process of each code and its contribution towards the MI spirit.

7. Acknowledgements

Research supported in part by NIH grant R01-AA018673 and NSF grant 1059095.

8. References

[1] W. Miller and S. Rollnick, *Motivational Interviewing: Preparing People for Change*. Guilford Press, 2002.

[2] W. R. Miller, T. B. Moyers, D. Ernst, and P. Amrhein, "Manual for the Motivational Interviewing Skill Code (MISC) Version 2.1." [Online]. Available: http://casaa.unm.edu/download/misc.pdf

[3] B. L. Burke, C. W. Dunn, D. C. Atkins, and J. Phelps, "The emerging evidence base for motivational interviewing: A meta-analytic and qualitative inquiry," *Journal of Cognitive Psychotherapy*, vol. 18, pp. 309–322, 2005.

[4] E. Proctor, H. Silmere, R. Raghavan, P. Hovmand, G. Aarons, A. Bunger, R. Griffey, and M. Hensley, "Outcomes for implementation research: conceptual distinctions, measurement challenges, and research agenda." *Adm Policy Ment Health*, vol. 38, no. 2, pp. 65–76, 2011.

[5] I. Androutsopoulos and P. Malakasiotis, "A survey of paraphrasing and textual entailment methods," *J. Artif. Int. Res.*, vol. 38, no. 1, pp. 135–187, May 2010.

[6] T. Moyers, T. Martin, D. Catley, K. J. Harris, and J. S. Ahluwalia, "Assessing the integrity of motivational interviewing interventions: reliability of the motivational interviewing skills code," *Behavioural and Cognitive Psychotherapy*, vol. 31, pp. 177–184, 2005.

[7] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. Van, and E. dykema Marie Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Computational Linguistics*, vol. 26, pp. 339–373, 2000.

[8] A. McCallum, D. Freitag, and F. C. N. Pereira, "Maximum entropy markov models for information extraction and segmentation," in *Proceedings of the ICML*, 2000, pp. 591–598.

[9] P. G. Georgiou, M. P. Black, A. Lammert, B. Baucom, and S. S. Narayanan, ""that's aggravating, very aggravating": Is it possible to classify behaviors in couple interactions using automatically derived lexical features?" in *Proceedings of Affective Computing and Intelligent Interaction (ACII), Lecture Notes in Computer Science*, Oct. 2011.