INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Exploiting Intra-annotator Rating Consistency through Copeland's Method for Estimation of Ground Truth Labels in Couples' Therapy

Karel Mundnich[1], *Md Nasir*[1,2], *Panayiotis Georgiou*[2], *Shrikanth Narayanan*[1]

[1]Signal Analysis and Interpretation Lab, University of Southern California, USA
[2]Signal Processing for Communication Understanding and Behavior Analysis Lab, University of Southern California, USA

{mundnich, mdnasir}@usc.edu, {georgiou, shri}@sipi.usc.edu

Abstract

Behavioral and mental health research and its clinical applications widely rely on quantifying human behavioral expressions. This often requires human-derived behavioral annotations, which tend to be noisy, especially when the psychological objects of interest are latent and subjective in nature. This paper focuses on exploiting multiple human annotations toward improving reliability of the ensemble decision, by creating a ranking of the evaluated objects. To create this ranking, we employ an adapted version of Copeland's counting method, which results in robust inter-annotator rankings and agreement. We use a simple mapping between the ranked objects and the scale of evaluation, which preserves the original distribution of ratings, based on maximum likelihood estimation. We apply the algorithm to ratings that lack a ground truth. Therefore, we assess our algorithm in two ways: (1) by corrupting the annotations with different distributions of noise, and computing the inter-annotator agreement between the ensemble estimates derived from the original and corrupted data using Krippendorff's α; and (2) by replacing one annotator at a time with the ensemble estimate. Our results suggest that the proposed method provides a robust alternative that suffers less from individual annotator preferences/biases and scale misuse.

Index Terms: Behavioral coding, annotation, annotator fusion, annotator ensemble, Copeland's counting method, inter-rater agreement.

1. Introduction

Interdisciplinary approaches that include advanced data analysis techniques have permeated into different areas of social sciences, as data becomes more accessible. As a consequence, data-driven approaches in understanding human behavior and emotion have become popular in recent years [1]. In clinical psychology and psychiatry, assessment of attributes describing human behavior is crucial for the improvement of the quality of therapy. Often known as 'behavioral codes', these behavioral descriptors are usually annotated by human experts based on the observation of audio-visual expressions or interactions/interviews of the subjects. While these behavioral codes are primarily used by psychology researchers, interdisciplinary efforts have attempted to automatically predict them from data using machine learning techniques in a supervised learning manner in order to create objective measures useful to track advances in diagnosis and therapeutic processes. For example, advances have been made in couples' therapy [2], addiction counseling [3], and analyzing behavior of children with autism spectrum disorder (ASD) [4] through machine learning in order to support the work of clinicians.

Current quantitative behavioral research mainly involves supervised learning tasks, where it is mandatory to include humans to produce perceptual labels on which classification/regression algorithms are trained. Nevertheless, annotation of the behavioral codes tends to be noisy due to bias and inconsistency of the annotators resulting from the latent and subjective nature of the codes [5, 6] and the complexity of the annotation task. To address these issues, often multiple annotators rate the same subjects. Fusion (or aggregation) of annotations is done in order to remove the effect of artifacts and outliers, leading to a more reliable estimation of the ground truth [5].

An active area of research is the problem of fusing annotations or ratings from different annotators to create an ensemble (estimate of the ground truth). Different techniques have been proposed. In [6], a noise and bias model for each annotator is proposed. By computing these parameters, annotators can be taken into a joint annotator space where their ratings are comparable. In [7] an Expectation-Maximization approach is used to find the latent ratings, while in [8], a similar approach is used to jointly model the latent ordinal labels and annotator reliability.

Recent advances in annotation research from the machine learning community suggest that ranking is better than rating [9, 10, 11]. Moreover, there have been interesting theoretical analyses of algorithms to create rankings [12, 13]. In [12], the authors prove that Copeland's counting method is optimal among Condorcet methods in the sense of achieving the information theoretic limits for recovering the top k-subset of a ranking. This property could impose recovery guarantees on the true labels if the annotation task is properly done, which serves as an inspiration to combine and use results of ranking theory into rated annotations.

This paper proposes a new annotation fusion method of ordinal ratings based on a modified version of Copeland's counting method [14], by considering ratings of objects as preferences of each annotator. This allows the combination of different scales of ratings based solely on preferences to come up with a unified vision of the order of the objects being rated, and without assuming any specific prior annotator models.

2. Data description

Our experiments in this paper are done on the ratings derived from Couples Therapy corpus of dyadic spoken interactions of a husband and wife in marital counseling [15] defined by the *Couples Interaction Rating System* (CIRS) [16, 17]. It has 13 different behavioral codes (see Table 1) and every code is rated on a scale from 1 to 9. The database contains a total of 5367 annotations that correspond to 1538 unique sessions from 186 couples rated by 17 trained annotators. Ratings include behavioral assessment of husband and wife along 13 behavioral codes. Each session is rated by a subset of the annotators ranging from 2 to 9 annotators, for a total of 5367 over 26146 possible ratings.

Copyright © 2017 ISCA

http://dx.doi.org/10.21437/Interspeech.2017-1599

3. Methodology

Our algorithm involves two steps to create a mapping between our rating space (1–9), a ranking space (the relative ranking between objects), and back: (1) using a modified version of Copeland's counting method to create a ranking of all the objects (sessions) being evaluated, and (2) creating a mapping from the ranking space back into the original rating space. This section describes both steps and the experiments executed to assess the performance of the algorithm.

3.1. Modified Copeland's counting method

Copeland's counting method is an algorithm to compute the underlying ranking of choices among different voters, where each one contributes with one vote. Each voter ranks objects by preference. For each pair of objects j and j', if j is higher than j' in the ranking, it gets assigned one vote. We repeat for all voters. If object j gets more votes than object j', j gets assigned one point. We repeat for all the remaining pairs of objects, and rank them according to the amount of points (score) obtained. Please refer to [18], pages 44–47 for details.

We introduce a variation of this method to create a fusion of ratings from annotators' ratings. Let $A \in \mathbb{R}^{m \times n}$ be an annotations matrix, where $m = |\mathcal{S}|$ ($\mathcal{S}$ is the set of sessions/objects) and $n = |\mathcal{A}|$ ($\mathcal{A}$ is the set of annotators). We consider that annotator i prefers session j over session j' if the rating of session j is greater than the rating of session j'. We then proceed in the same manner as Copeland's counting method (Algorithm 1).

The assumptions that we consider are the following: (1) annotators rate high behavioral code scores with high ratings and low behavioral code scores with low ratings; (2) each behavioral code is treated independently; and (3) annotators do not use values above or below the maximum or minimum values of the given rating scale.

The proposed algorithm is designed for mostly complete annotation matrices, where there is a small amount of missing values. Given that our ratings matrices are sparse, we perform the analysis by imputing the missing values with the average rating for each of the rated objects. This prevents spurious high scores coming from annotators with low amounts of annotations.

3.2. Rankings to ratings

Algorithm 1 outputs a vector containing the scores assigned to each object (session) being rated. A direct mapping from scores into ratings from 1–9 is achieved through an affine transformation between the ranking space and the rating space. Nevertheless, this transformation creates a distribution of ratings that does not represent the structure of behavioral code scores for the set of analyzed sessions. To work around this issue, we perform an estimation of the density of ratings of the annotations matrix A by computing the histogram of the ratings. This computation preserves the distribution of ratings from matrix A, allowing a mapping between the ranking space and the ratings space that also preserves these statistics.

The output of Algorithm1 is used to rank sessions by their scores. To create the mapping between ranking space and rating space, we compute the histogram counts $c_1, \ldots, c_9$ of the ratings matrix A and the proportions $\rho_1, \ldots, \rho_9$ for each possible value in the scale, where $\rho_l = c_l / \Sigma_{l'} c_{l'}$. In the sorted scores vector, we assign (from lowest to highest ranked) a value of 1 to the first $m \cdot \rho_1$ elements, a value of 2 to the following $m \cdot \rho_2$. We proceed until we have assigned rating values to all the ranked sessions. As previously mentioned, this mapping preserves the statistics of the collected ratings, while considering the underlying preference among all raters.

Algorithm 1: Modified Copeland's counting method

input : $A \in \mathbb{R}^{m \times n}$: Annotations matrix
m, n: # of sessions, # of annotators
output: Scores vector $\in \mathbb{R}^m$

```
scores ← zeros(m,1);
for j ← 1 to m − 1 do
    for j' ← i + 1 to m do
        votes_j ← 0;
        votes_j' ← 0;
        for i ← 1 to n do
            rating_j ← A[j, i];
            rating_j' ← A[j', i];
            if rating_j > rating_j' then
                votes_j ← votes_j + 1;
            else if rating_j < rating_j' then
                votes_j' ← votes_j' + 1;
        if votes_j > votes_j' then
            scores[j] ← scores[j] + 1;
        else if votes_j < votes_j' then
            scores[j'] ← scores[j'] + 1;
        else
            scores[j] ← scores[j] + 1/2;
            scores[j'] ← scores[j'] + 1/2
```

3.3. Assessment of the algorithm

There is an intrinsic difficulty in assessing the performance of the algorithm, as there is no ground truth to compare the results to within our data set. Even more, since our algorithm goes from a rating space into a ranking space and back into a rating space, assessment depends on both stages, and not only the output (ensemble). To work around these difficulties, we propose three different experiments to assess its performance: (1) corrupt the initial ratings with noise, and compare the original ensemble with the ensembles from the corrupted data; (2) perform cross validation by leaving one annotator out, and compute Krippendorff's α [19, 20] over the ratings matrix including the ensemble; and (3) compute the consistency of the output ensembles by leaving one annotator out. In all three experiments, we compare the results of the proposed modified Copeland's ensembles with the ensembles obtained by the average ratings.

3.3.1. Corruption of the ratings with noise

To investigate the stability of the algorithm against noisy inputs, we have corrupted all the ratings r_{ijk} with the following rule:

$$r'_{ijk} = r_{ijk} + w_{ijk}, \tag{1}$$

where

$$w_{ijk} = \begin{cases} -1, & \text{w.p.} \quad p, \\ 0, & \text{w.p.} \quad 1 - 2p, \\ +1, & \text{w.p.} \quad p, \end{cases} \tag{2}$$

and $i \in \mathcal{A}$, $j \in \mathcal{S}$, $k \in \mathcal{C}$ (behavioral codes set), and $p \in \{1/10, \ldots, 1/3\}$.

The comparisons have been performed in the following manner: For each code k, annotation matrix A_k has been passed through our algorithm, obtaining an ensemble vector $\boldsymbol{e}_k$. Then, the annotations matrix is corrupted with noise, obtaining A'_k and the corresponding ensemble vector $\boldsymbol{e}'_k$. Krippendorff's α is computed between $\boldsymbol{e}_k$ and $\boldsymbol{e}'_k$. The corruption process is repeated throughout 10 iterations, and the average and standard

deviation Krippendorff's α is computed. This process is repeated for each value of p, for each different behavioral code.

3.3.2. Leave-one-out average agreement

For each behavioral code k, we leave one annotator out and produce an annotation ensemble $\boldsymbol{e}_k^{\mathcal{A}\cap\{i\}^c}$. We replace the left-out annotations with the ensemble annotations, and compute the inter-annotator agreement of this replaced-leave-one-out matrix

$$A_k^{\mathcal{A}\cap\{i\}^c} = [\boldsymbol{a}_k^1|\cdots|\boldsymbol{e}_k^{\mathcal{A}\cap\{i\}^c}|\cdots|\boldsymbol{a}_k^n]. \quad (3)$$

We do this for each of the $n = 17$ annotators and compute the mean α. This mean agreement gives us an idea of how much agreement an ensemble creates within the annotations matrix, but does not tell us how close we are to the ground truth.

3.3.3. Leave-one-out ensemble agreement

In our last assessment routine, we consider all the ensembles $\boldsymbol{e}_k^{\mathcal{A}\cap\{i\}^c}$ calculated in the previous experiment to build an ensemble matrix E_k:

$$E_k = [\boldsymbol{e}_k^{\mathcal{A}\cap\{1\}^c}|\cdots|\boldsymbol{e}_k^{\mathcal{A}\cap\{i\}^c}|\cdots|\boldsymbol{e}_k^{\mathcal{A}\cap\{n\}^c}]. \quad (4)$$

To understand how robust these ensembles are with respect to leaving one annotator out, we compute Krippendorff's α for this ensembles matrix.

4. Results

4.1. Stability from data corruption

Figure 1 shows the mean of Krippendorff's α between $\boldsymbol{e}_k$ and $\boldsymbol{e}'_k$ for 10 iterations of each corruption process together with the standard deviation. A horizontal black dashed line has been plotted for reference as good inter-annotator agreement is considered for $\alpha \geq 0.8$ [20]. Figure 1a shows the stability analysis under noise for our proposed method. We observe two different clusters of behavioral codes: (1) the top six behavioral codes where inter-annotator agreement is not very affected by noise corruption, and where most agreement is above $\alpha = 0.9$; and (2) a group of behavioral codes that is more affected by noise, where most values go below $\alpha = 0.8$ for a values of $p = 1/3$ (Equation 2).

Figure 1b shows the same results for the average ensembles. We observe a similar trend for the top behavioral codes, even achieving higher values for α. Nevertheless, performance overall decays faster for each behavioral codes, having lines going below $\alpha = 0.8$ from $p = 1/7$ onwards. For the lower cluster of behavioral codes, all of them go below $\alpha = 0.8$ for the lower behavioral codes' cluster.

To investigate the possible reasons for the differences in performance, we have plotted the histograms of ratings for each behavioral code in Figure 2. Behavioral codes with values that are highly skewed towards a rating of 1 (such as "solicits partner's perspective", "clearly defines problem", "offers solutions", "pressures for change", and "withdraws") are not stable under slight variations of the ratings, while behavioral codes with good relative frequency for each rating in the scale excel under perturbations.

Relative higher stability for behavioral codes suggest that our proposed algorithm is more robust against perturbations, meaning that it can handle noisy inputs from annotators and better find the underlying structure of the data.

4.2. Leave-one-out average agreement

Table 1 shows the results for the leave-one-out experiment cross validation. The results obtained show that both methods perform very similarly. For 5 out of 13 behavioral codes, $\bar{\alpha}_{\text{Copeland}}$ shows a statistically significant increase ($p < 0.05$) over $\bar{\alpha}_{\text{mean}}$, while on 8 out of 13, we cannot reject that $\bar{\alpha}_{\text{mean}}$ and $\bar{\alpha}_{\text{Copeland}}$ are the same. This implies that replacing our ensemble with each annotator produces more inter-annotator agreement than human annotators alone, and it does at least as well as considering the mean of the ratings.

Table 1: *Inter-annotator agreement for all behavioral codes. α: Krippendorff's α for the original annotations; $\bar{\alpha}_{mean}$, $\bar{\alpha}_{Copeland}$: Average Krippendorff's α computed by leaving one annotator out and replacing it with the $n-1$ annotators' ensemble († indicates statistically significantly higher than $\bar{\alpha}_{mean}$, $p < 0.05$).*

Behavioral code	α	$\bar{\alpha}_{\text{mean}}$	$\bar{\alpha}_{\text{Copeland}}$
acceptance of the other	0.6279	0.7056	0.7103
blame	0.6364	0.7106	0.7189
responsibility for self	0.4785	0.5745	0.5855†
solicits perspective	0.3497	0.4635	0.4646
states external origins	0.3764	0.4928	0.5092†
discussion	0.6207	0.6943	0.7050†
clearly defines problem	0.4229	0.5239	0.5267
offers solutions	0.5474	0.6280	0.6293
negotiates	0.7010	0.7612	0.7659
makes agreements	0.5941	0.6744	0.6860†
pressures for change	0.3939	0.4978	0.4996
withdraws	0.4072	0.5068	0.5093
avoidance	0.4121	0.5198	0.5325†

Table 2: *Krippendorff's α for ensemble matrices E_k designed based on mean and Copeland ensembles in a leave-one-annotator-out setup (Equation 4).*

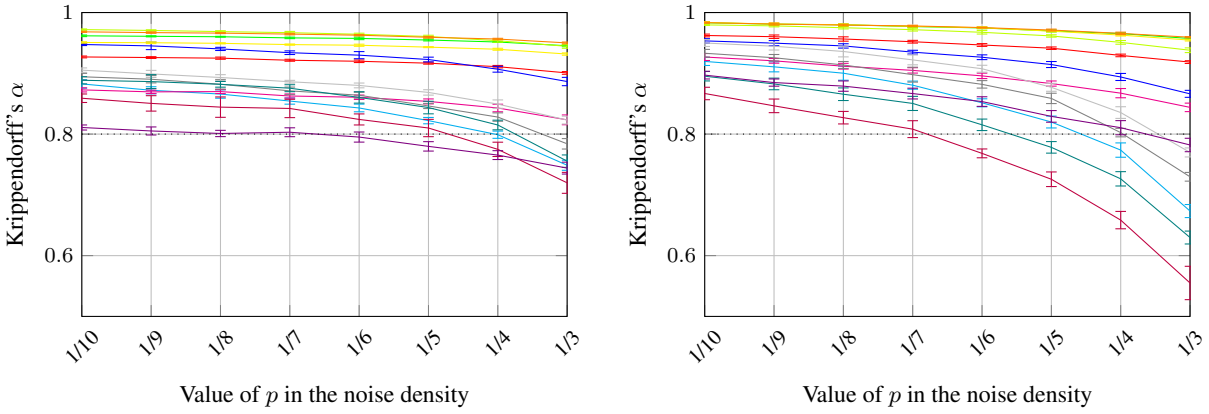
Behavioral code	α_{mean}	α_{Copeland}
acceptance of the other	0.9730	0.9773
blame	0.9800	0.9857
responsibility for self	0.9618	0.9795
solicits perspective	0.9498	0.9631
states external origins	0.9435	0.9737
discussion	0.9796	0.9875
clearly defines problem	0.9569	0.9626
offers solutions	0.9699	0.9643
negotiates	0.9836	0.9866
makes agreements	0.9791	0.9862
pressures for change	0.9317	0.9633
withdraws	0.9394	0.9641
avoidance	0.9368	0.9694

4.3. Leave-one-out ensemble agreement

Results for the ensemble robustness leaving one annotator experiment are shown in Table 2. Our proposed algorithm has greater ensemble agreement in 12 of the 13 rated behavioral codes. This means that our algorithm is able to extract more information from the group's overall agreement than what the mean among ratings is able to capture, implying lower variability coming from each individual annotator.

5. Discussion

The results suggest that the proposed algorithm is capable of finding finer underlying structure within the data than creating an annotator ensemble based on averaging ratings. Although exact assessment of the algorithm is challenging due to the high quantity of missing values in the present data set and a lack



(a) *Modified Copeland's ensemble (proposed).* (b) *Mean ensemble (baseline).*

Figure 1: *Data corruption stability analysis. Each rating has been corrupted by a noise, where p represents the probability of the original rating to be increased/decreased by 1. The plotted values are the inter-rater agreement between the original rating's ensemble and the corrupted rating's ensemble averaged over 10 iterations. The error bars correspond to one standard deviation. Note: No noise corruption implies a Krippendorff's α of 1 for every behavioral code. Legend: acceptance of the other, blame, responsibility for self, solicits perspective, states external origins, discussion, clearly defines problem, offers solutions, negotiates, makes agreements, pressures for change, withdraws, avoidance.*

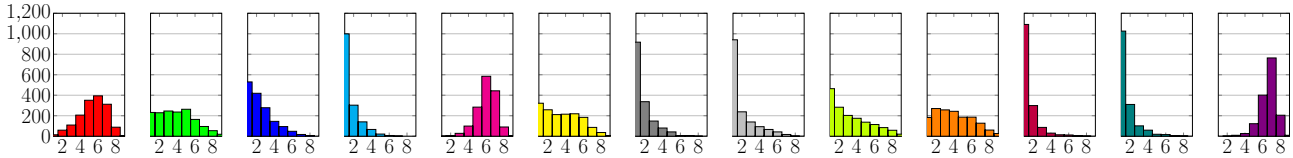


Figure 2: *Rating distributions for each annotated behavioral code. The y-axis represents relative counting frequency, while the x-axis represents behavioral codes ratings (scale is 1 to 9). Histograms' colors refer to behavioral codes as defined in the legend of Figure 1.*

of ground truth, the experiments imply more robustness to locally changing values due to annotation noise/interpretation and higher robustness with respect to missing annotators in the proposed algorithm compared to the average ensemble.

Poor results for 6 out of 13 behavioral codes in the stability experiment suggest intrinsic problems in the distribution of behavioral codes (Figure 2). Observing Figures 1 and 2 in parallel show that low robustness to noise is directly related to highly skewed rating distributions, where high values are poorly represented. When this occurs, both algorithms lack robustness. Nevertheless, this is more prominent in the average ensemble. Our algorithm suffers from this condition as missing low ratings are considered a win over missing values, needing to impute missing values. This issue suggests the exploration of other methods to impute missing data.

The proposed method forces a mapping from ranking space (lead by points from Copeland's method) into rating space. This let us assess our algorithm in comparison to an average baseline, while having an output in the original rating space. Nevertheless, this mapping forces a change in the distribution of scores, which is often uniform-looking (figure not included). Our mapping is a simple solution to this complex problem, but needs revision as it may be a suboptimal solution to this problem, which has the potential to affect the estimation of the underlying ground truth.

From a classification/regression perspective, working in the ranking space offers more resolution and a close-to-uniform distribution of points, which may lead to improvements in finding audio-visual features that are highly correlated with each one of the behavioral codes, as well as improvements in the overall classification/regression performance.

Finally, our algorithm partially finds a solution to the question: How to compare ratings from different annotators? While directly comparing the ratings of different annotators is deemed as a highly complex task, comparing their preferences and weighing them seems a natural approach for annotator fusion. This permits the fusion among ratings within different scale uses (or scale misuse) among different annotators, while providing higher resolution results in the ranking space.

6. Conclusions

The present paper presents and explores an annotator fusion algorithm based on an adaptation of Copeland's counting method by reinterpreting voters' choices as preferences. This algorithm is presented to support the use of audio-visual machine learning techniques in the quantification of human behavior and behavioral expressions, as a means to support clinicians and psychology researchers in the study of latent and subjective objects.

The proposed algorithm has been implemented and its performance tested in a real-world couples' therapy data set, where annotators have annotated the scores of 13 different behavioral codes that are pertinent to couples' therapy, such as acceptance of other and blame, among others.

The results suggest that the method is able to recover fine structure from the underlying latent state of the behavioral code, by being more robust to noise and by better capturing the group consensus. Nevertheless, the algorithm needs data imputation for success, an area that needs further exploration in the ratings domain.

As part of our future work, we will focus on improving the mapping procedure from the ranking to the rating space and employ recent matrix completion techniques for testing our approach on data sets with many missing values. As proposed changes to the modified version of Copeland's counting method, different weights for different annotators according to their own reliabilities must be considered.

7. References

[1] S. Narayanan and P. G. Georgiou, "Behavioral signal processing: Deriving human behavioral informatics from speech and language," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1203–1233, 2013.

[2] M. Black, A. Katsamanis, C.-C. Lee, A. C. Lammert, B. R. Baucom, A. Christensen, P. G. Georgiou, and S. S. Narayanan, "Automatic classification of married couples' behavior using audio features." in *INTERSPEECH*, 2010, pp. 2030–2033.

[3] B. Xiao, C. Huang, Z. E. Imel, D. C. Atkins, P. Georgiou, and S. S. Narayanan, "A technology prototype system for rating therapist empathy from audio recordings in addiction counseling," *PeerJ Computer Science*, vol. 2, p. e59, 2016.

[4] D. Bone, C.-C. Lee, M. P. Black, M. E. Williams, S. Lee, P. Levitt, and S. Narayanan, "The psychologist as an interlocutor in autism spectrum disorder assessment: Insights from a study of spontaneous prosody," *Journal of Speech, Language, and Hearing Research*, vol. 57, no. 4, pp. 1162–1177, 2014.

[5] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *Journal of Machine Learning Research*, vol. 11, no. Apr, pp. 1297–1322, 2010.

[6] M. Nasir, B. Caycom, P. Georgiou, and S. Narayanan, "Redundancy analysis of behavioral coding for couples therapy and improved estimatino of behavior from noisy annotations," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, 2015, pp. 1886–1890.

[7] A. Ramakrishna, R. Gupta, R. B. Grossman, and S. S. Narayanan, "An expectation maximization approach to joint modeling of multidimensional ratings derived from multiple annotators," in *Interspeech 2016*, 2016, pp. 1555–1559. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2016-270

[8] K. Audhkhasi and S. Narayanan, "A globally-variant locally-constant model for fusion of labels from multiple diverse experts without using reference labels," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 4, pp. 769–783, 2013.

[9] H. P. Martinez, G. N. Yannakakis, and J. Hallam, "Don't classify ratings of affect; rank them!" *IEEE Transactions on Affective Computing*, vol. 5, no. 3, pp. 314–326, 2014.

[10] G. N. Yannakakis and H. P. Martinez, "Grounding truth via ordinal annotation," in *Affective Computing and Intelligent Interaction (ACII), 2015 International Conference on*. IEEE, 2015, pp. 574–580.

[11] G. N. Yannakakis and H. P. Martnez, "Ratings are overrated!" *Frontiers in ICT*, vol. 2, p. 13, 2015.

[12] N. Shah and M. J. Wainwright, "Simple, Robust and Optimal Ranking from Pairwise Comparisons," *arXiv:1512.08949*, 2015.

[13] N. B. Shah, S. Balakrishnan, J. Bradley, A. Parekh, K. Ramchandran, and M. J. Wainwright, "Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence," *Journal of Machine Learning Research*, vol. 17, no. 58, pp. 1–47, 2016.

[14] H. Nurmi, "Voting Procedures: A Summary Analysis," *British Journal of Political Science*, vol. 2, pp. 181–208, 1983.

[15] A. Christensen, D. C. Atkins, S. Berns, J. Wheeler, D. H. Baucom, and L. E. Simpson, "Traditional versus integrative behavioral couple therapy for significantly and chronically distressed married couples," *Journal of consulting and clinical psychology*, vol. 72, no. 2, p. 176, 2004.

[16] J. Jones and A. Christensen, "Couples interaction study: Social support interaction rating system," University of California, Los Angeles, Tech. Rep., 1998.

[17] C. Heavey, D. Gill, and A. Christensen, "Couples interaction rating system 2 (CIRS2)," University of California, Los Angeles, Tech. Rep., 2002.

[18] D. Lippman, *Math in Society*. Pierce College Ft Steilacoom, 2013.

[19] K. Krippendorff, "Estimating the reliability, systematic error and random error of interval data," *Educational and Psychological Measurement*, vol. 30, no. 1, pp. 61–70, 1970.

[20] ——, *Content analysis: An introduction to its methodology*. Thousand Oaks, CA: Sage, 2004.