# Predicting Group Performance Using Machine Learning Models

Fantillo Joshua
School Of Computing
*The University of the Fraser Valley*
Abbotsford, Canada
joshua.fantillo@student.ufv.ca

***Abstract***

This project further addresses the issue of poor group performance. By using different machine learning models, we try to predict the group performance of certain groups in the GAP corpus. After doing multimodal analysis of the group data and running the different machine learning models we were able to get an accurate result for predicting how well a group would preform on a task together based on the linguistics, and speech features of a group.

***Keywords***

Group interaction, group performance, multimodal analysis, data augmentation, data modification, semi-supervised learning, machine learning models

## I. INTRODUCTION

A big part of working in a company is team or group projects and assignments. This for the most part could be an issue as not all people work best with each other as they do with other people. To optimize this performance then there should be a way to optimize the formation of teams. If we optimize the pairing of groups, then we can optimize the performance of the workers.

In this project we try to successfully determine how well a group of people will work together based on a variety of speech and linguistic features. Doing this will help us create a team that works best together for dealing with other group tasks.

The project goals were to analyze the relationship between speech features and linguistic features in a small group interaction to predict group performance, to implement machine learning models for group interaction, and to assess the machine learning algorithms on performance.

We achieve these goals by extracting speech features and linguistic features from the GAP corpus dataset. By extracting these features and performing analysis on the data and modifying and cleaning it we should be able to predict a group's performance.

## II. RELATED WORK

Understanding group performance has always been a concern withing the field of psychology. Just more recently has that field brought in computational methods to help with that understanding.

Some research has been done on this automatic prediction of group performance before. Most of this project has been following and building upon 'Predicting Group Performance in Task-based Interaction' [1] by Gabriel Murray and Catharine Oertel. This project generally follows what Murray and Oertel did as well as tests other features and modifications to see the results they produce.

## III. MULTIMODAL ANALYSIS OF GROUP PERFORMANCE

In this section we describe different features we extracted from the corpus and how we did it.

### *A. Speech Features*

To extract some speech features we used the python_speech_features library [2]. Using this we were able to extract the Mel-frequency cepstrum coefficient (MFCC), the Mel-filterbank (fbank), the log Mel-filterbank (logfbank), and the spectral subband centroid (SSC) from the dataset. We got these features for every sentence in the dataset, taking the average of each group and implementing it into our model.

### *B. Linguistic Features*

The following linguistic features were extracted from the manual transcript in the corpus.

**Parsed Sentences:** All sentences are parsed using the NLTK word tokenizer [3]. Doing this we extract some features like getting the total number of words used in each group and in the whole corpus as well as total number of sentences used. We also got the average amount of words used per sentence.

**Filled Pause:** Found the number of filled pauses, 'uh' or 'um', for each group and got rid of empty sentences in the parsed sentences.

**Sentiment Scores:** We got the sentiment scores by using textblob [4]. From polarity scores textblob gave we were able to assign a positive, neutral, and negative value to each sentence.

**Stop Words:** We further cleaned the words by removing stop words from each sentence in the dataset.

**Lemmatization:** We lemmatized the dataset by using NLTK WordNetLemmatizer [5]. This was done so we could simplify the sentence's words into their root words to accurately get frequencies of the words.

**Cosine Similarity:** We used sklearn's cosine_similarity [6] to get each sentences similarity with the sentences

directly in front and behind of it. After getting each sentences similarity we took the average similarity of all the sentences in each group.

**Parts Of Speech Tags:** We got the POS Tags for each word in every sentence. To do this we used NLTK's pos_tag [7].

**Word and Tag Frequency:** We got each word and tags frequency for each group's transcript using the NLTK's FreqDist [8]. By doing this we were able to get the top 100 most common words for each group and top 15 most common tags.

**Bag of Words and Tags:** From the data we got from word and tag frequencies we were able to get our Bag of Words and Bag of Tags using the most common words (100) and tags (15).

**Type Ratio:** We got the type of token ratio and the type tag ratio for each group in the dataset.

### C. *Facial Features*

The facial feature extraction portion of this experiment was abandoned as it was not possible to accurately extract the facial expressions from the videos in the data set. Only sometimes would it detect the facial expressions of an individual but most of the time (more than 95%) it would not detect the individuals face. This was mostly due to the lighting and the poor quality of the video.

## IV. EXPERIMENTAL SETUP

### A. *Corpus*

The corpus this experiment was run on was the UFV's Group Affect and Performance (GAP) Corpus [9]. This corpus got individuals into a group of size 2 to 4 to complete a winter survival task. The groups were put into a hypothetical situation that their plane crashed, and they salvaged 15 items from the crash. From those 15 items they needed to rank them in order of most useful to least useful.

The corpus has many features from both the groups and individuals. Some of these features were the transcripts, audio files, video files, as well as scores for certain aspects of the group tasks and individual tasks.

These scores were the average group score (AVG), group time expectation (Group_TE), group worked well together (Group_WW), group time management (Group_TM), group efficiency (Group_Eff), group quality of work (Group_QW), and group overall satisfaction with meeting (Group_Sat). All these scores were calculated and saved as well as the same scores for each individual person.

The reason this dataset was chosen is that the set has a well-defined task that ranks both individual and group scores. This can give us clear measurements of group performance

In this corpus there are 28 meetings. All these meetings are in English, and all have between 2 to 4 people in the groups. Because the purpose of our study is to see if we can predict group performance the dataset with this corpus is quite small. It should be noted that no secondary data source was used.

### B. *Machine Learning Models*

We use 3 machine learning models in this experiment. We use Random Forests (RF), a Decision Tree Regressor (DTR) and the K Nearest Neighbor (KNN) model. We mostly focus on comparing RF and DTR. For each of these models we used different values and different modifications of the dataset we found.

For the RF we used different values of the n estimator (1, 5, 8, 20, 50), for the DTR we used different values for the max depth (1, 2, 3, 5, 10), and for the KNN we used different values for the n neighbors (1, 2, 5, 8, 10). It should also be noted that when we were comparing RF vs DTR we also changed the number of estimators (1, 2, 5, 10, 50). We use 21 slightly different modifications (Table 1) of the dataset we created to see which one would give more accurate results or which data we found was either not relevant to the study or not accurately calculated.

### C. *Evaluation*

For evaluating the models, we got each of the model's accuracy. For comparing the RF and DTR we got the negative mean absolute error for both and compared accordingly.

| Modification | Dropped Data |
|---|---|
| Modification 1 | Drop Total Words |
| Modification 2 | Drop Total Sentences |
| Modification 3 | Drop Neutral Sentiment |
| Modification 4 | Drop Filled Pauses |
| Modification 5 | Drop Cosine Similarity |
| Modification 6 | Drop Type Token Ratio |
| Modification 7 | Drop Type Tag Ratio |
| Modification 8 | Drop MFCC |
| Modification 9 | Drop log fBank |
| Modification 10 | Drop SSC |
| Modification 11 | Drop Avg Year |
| Modification 12 | Drop English |
| Modification 13 | Drop Group_TE |
| Modification 14 | Drop Group_WW |
| Modification 15 | Drop Group_TM |
| Modification 16 | Drop Group_Eff |
| Modification 17 | Drop Group_Sat |
| Modification 18 | Drop Mod1+ Mod2 |
| Modification 19 | Drop Mod6 + Mod7 |
| Modification 20 | Drop Mod (8 + 9 + 10) |
| Modification 21 | Drop Mod 11-17 |

Table 1: Modification of the Data Set

## V. RESULTS

The results varied quite a bit depending on what data modification we used and what machine learning model we used. The results ranged from 55.6% (using the KNN=8) accuracy to 91.3% the (using DTR depth = 10) accuracy. It should be noted that the highest accuracy model on average was the DTR model with depth=10,

with an average accuracy of 86.9%. It should also be noted that the highest dataset modification accuracy was the 15th modification dataset with an average accuracy of 78.0%. Looking at table 2 we can see the highest accuracy and lowest accuracy for each machine learning model used.

Although the modified dataset with the highest accuracy is the 15th modified dataset, the dataset that gives us the highest accuracy overall for a specific machine learning model is the 20th dataset.

When comparing the DTR and RF using the negative absolute error we can see that the DTR performs better than RF in almost every single modification and scenario. Although DTR performs better on average than the RF, both the RF and DTR perform much better than the KNN.

| | KNN | DTR | RF |
|---|---|---|---|
| Model | KNN = 1 | Depth = 10 | N Est = 9 |
| Highest | 80% | 91.3% | 89.5% |
| Model | KNN = 8 | Depth = 1 | N Est = 1 |
| Lowest | 55.6% | 76.1% | 62.3% |

Table 2

## VI. Discussion

It was found that KNN wasn't very accurate, unless its number of neighbors were equal to 1. This is mostly because there may be too many features in the dataset, we used such that the KNN model isn't as accurate as DTR and RF.

Both DTR and RF were very accurate for the most part. Both averaged an average accuracy of around 85% to 86% except in the cases where the N estimator was 1 and when the depth in the DTR was 1.

In further experiments I would like to find a way to implement emotional face detection. This was tried for this project, but it couldn't be implemented as reading the emotions off the faces of the individuals would almost never register. This is due to either bad lighting or videos being too pixelated. This can be built upon by either building a more sophisticated emotion detector or by using another dataset with better lighting/better video quality.

There were certain tools like openface, opencv, and emoPy that were all considered and implemented for this project to read emotions of the individuals faces, but none of them succeeded at successfully reading emotions. Hence why the emotion detection was dropped from the project.

## VII. Conclusion

We have produced results that show we can predict the group performance on the winter survival task with up to 91.3% accuracy. This was done using multimodal features of the group's interaction together. In total we used 4 speech features as well as multiple linguistic features. Emotional facial features were considered in this project as well but were not implemented as extracting emotional facial features from the dataset wasn't possible.

Using both the speech features and linguistic features it was found that just using the linguistic features on their own produce very good results. We also noticed that the highest accuracy was found using only linguistic features and not speech features. It should be noted that this could have been caused by the data not having a very big pool of speech features used.

## Acknowledgment

Joshua Fantillo is a student at The University of the Fraser Valley completing a capstone project.

## REFERENCES


[1] Gabriel Murray and Catharine Oertel. 2018. Predicting Group Performance in Task-Based Interaction. In 2018 International Conference on Multimodal Interaction (ICMI '18), October 16–20, 2018, Boulder, CO, USA. ACM, New York, NY, USA, 7 pages.

[2] python_speech_features 0.4, (2021), Retrieved October 29, 2021 from https://pypi.org/project/python_speech_features/0.4/

[3] nltk.tokenize package, (Oct 19, 2021), Retrieved October 25 2021 from https://www.nltk.org/api/nltk.tokenize.html

[4] TextBlob, (2020), Retrieved October 25, 2021 from https://textblob.readthedocs.io/en/dev/quickstart.html

[5] nltk.stem.wordnet, (Oct 19, 2021), Retrieved October 25 2021 from https://www.nltk.org/_modules/nltk/stem/wordnet.html

[6] scikit learn, (2007-2021), Retrieved October 26, 2021 from https://scikitlearn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html

[7] Categorizing and Tagging Words, (Sept, 4, 2019), Retrieved October 26, 2021 from https://www.nltk.org/book/ch05.html

[8] nltk.probability module, (Oct 19, 2021), Retrieved October 29, 2021 from https://www.nltk.org/api/nltk.probability.html

[9] Gabriel Murray, McKenzie Braley, The GAP Corpus, (2019), Retrieved October 15, 2021 from https://sites.google.com/view/gap-corpus/home