IMPROVING SWARM PERFORMANCE BY APPLYING MACHINE
LEARNING TO A NEW DYNAMIC SURVEY

A Thesis
presented to
the Faculty of California Polytechnic State University,
San Luis Obispo

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Electrical Engineering

by
John Taylor Jackson

May 2018

© 2018

John Taylor Jackson

ALL RIGHTS RESERVED

COMMITTEE MEMBERSHIP

TITLE: Improving Swarm Performance by Applying
Machine Learning to a New Dynamic Survey

AUTHOR: John Taylor Jackson

DATE SUBMITTED: May 2018

COMMITTEE CHAIR: Franz Kurfess, Ph.D
Professor of Computer Science

COMMITTEE MEMBER: Tina Smilkstein, Ph.D
Professor of Electrical Engineering

COMMITTEE MEMBER: Jane Zhang, Ph.D
Professor of Electrical Engineering

ABSTRACT

Improving Swarm Performance by Applying Machine Learning to a New Dynamic Survey

John Taylor Jackson

A company, Unanimous AI, has created a software platform that allows individuals to come together as a group or a human swarm to make decisions. These human swarms amplify the decision-making capabilities of both the individuals and the group. One way Unanimous AI increases the swarm's collective decision-making capabilities is by limiting the swarm to more informed individuals on the given topic. The previous way Unanimous AI selected users to enter the swarm was improved upon by a new methodology that is detailed in this study. This new methodology implements a new type of survey that collects data that is more indicative of a user's knowledge on the subject than the previous survey. This study also identifies better metrics for predicting each user's performance when predicting Major League Baseball game outcomes throughout a given week. This study demonstrates that the new machine learning models and data extraction schemes are approximately 12% more accurate than the currently implemented methods at predicting user performance. Finally, this study shows how predicting a user's performance based purely on their inputs can increase the average performance of a group by limiting the group to the top predicted performers. This study shows that by limiting the group to the top predicted performers across five different weeks of MLB predictions, the average group performance was increased up to 5.5%, making this a superior method.

TABLE OF CONTENTS

	Page
LIST OF TABLES	vi
LIST OF FIGURES	vii
CHAPTER	
1. INTRODUCTION	1
2. BACKGROUND	3
2.1 Swarm Intelligence	3
2.2 Group Psychology	3
2.3 Unanimous AI and Unu	5
2.4: Current Surveys	8
2.4.1: Using Surveys to Weed Out Unu Participants	8
2.4.2 Using Surveys to Create Single Collective Predictions.....	14
3. SYSTEM DESIGN	17
3.1 Initial Concept.....	17
3.1.1 Dynamic Survey	17
3.1.2 Machine Learning	20
3.2 Dynamic Survey Requirements	21
4. SYSTEM IMPLEMENTATION	22
4.1 Dynamic Survey First Prototype	22
4.2 Dynamic Survey First Prototype Feedback	32
4.3 Static Survey Overview	33
4.4 First Dynamic Survey Prototype Findings.....	38
4.5 Second Dynamic Survey Prototype	38
4.6 MLB Static Survey.....	43
4.7 Final Dynamic Survey Iteration	47
4.8 Machine Learning on Dynamic Survey Data	53
4.9 Data Analysis	62
4.10 Predicting User's Performance With Machine Learning	76
5. CONCLUSION	93
6. FUTURE WORK	95
BIBLIOGRAPHY	96

LIST OF TABLES

Table	Page
1. Types of Data Gathered for Each Match	55
2. Types of Data Gathered for Each User	58
3. Average User Performance for Initial Predictions and Final Predictions and Vegas Performance	61
4. Machine Learning Model and Performance for the Kernel Densities for Support Scores as Input	77
5. Machine Learning Models and Performance Using All Data Collected and Calculated From This Study	78
6. Machine Learning Models and Performance Using All Data Besides Outlier and Support Scores	79
7. Machine Learning Models and Performance Using Most Correlated Data	80

LIST OF FIGURES

FIGURE	Page
1. Unanimous AI's Process of Weeding Out Users	1
2. Example of a Social Swarm Being Used to Predict the Outcome of the Kentucky Derby	7
3. Example Questions for a EPL Game Prediction	10
4. Example Questions for All EPL Games On a Given Week	11
5. Example Metadata Questions for the EPL Survey	12
6. Example of a Swarm Being Used to Predict the Outcome of a MLB Game Between the Chicago White Sox and the Boston Red Sox.....	14
7. Thought Process While a User Is In a Social Swarm	15
8. Thought Process While a User Takes a Survey.....	16
9. Example of an Initial Dynamic Survey Question and Slider	18
10. Example of an User's Initial Estimate to the Initial Dynamic Survey Question and Slider Model.....	19
11. Example of an User's Initial Estimate and Application Perturbation on the Initial Dynamic Survey Question and Slider Model.....	19
12. Example of an User's Final Estimate After the Application's Perturbation on the Initial Dynamic Survey Question and Slider Model.....	20
13. Diagram of the Data Flow of the Web Application.....	22
14. Screenshot of the Initial Login Screen.....	23
15. Screenshot of the Instructions Screen.....	24
16. Screenshot of an example NBA First Prediction Screen a user would have seen	25
17. Example of a User's Second Prediction Given an "Expert's Prediction"	25
18. User's Thought Process Once Exposed to the "Expert's Prediction"	26
19. Screenshot of an Example Probabilistic Wager Slider for a Given Game	26
20. Screenshot of an Example Summary of Predictions Table	29
21. Screenshot of a Favored Team Questionnaire	29
22. Screenshot of a Favored Team Questionnaire	30
23. Summary of User's Navigation Through the Initial Prototype	31
24. Example Data Saved From a User Predicting 8 Games One Week.....	32
25. Example Static Survey Sign In Screen.....	34
26. Example Static Survey Instructions Screen	34
27. Example Static Survey Questions for Predicting an NBA Game	35
28. Example Static Survey Questions for Users Predictions of the Week Out of Every Game	36

29. Example Static Survey Questions for Users Predictions of the Week Out of Every Game	37
30. User Sign In Page	39
31. User Instructions Page	40
32. Second Prototype Slider Screen Before a User's Prediction	41
33. Second Prototype Slider Screen With a User's Prediction	41
34. Second Prototype Slider Screen After Crowd Prediction Perturbation	42
35. User Instructions for MLB Static Survey	43
36. Worker ID and Metadata User Sign In Page	44
37. Example MLB Game Survey Each User Completed	45
38. Example MLB Game Survey Each User Completed	46
39. Example MLB Game Survey Each User Completed	47
40. Example MLB Dynamic Survey Sign In Page Before Completion	48
41. Example MLB Dynamic Survey Sign In Page After Completion	48
42. Example MLB Dynamic Survey Instructions Page	49
43. Example MLB Dynamic Survey Initial User's Prediction For a Game	49
44. Example MLB Dynamic Survey Final User's Prediction For a Game	50
45. Example MLB Dynamic Survey Summary of User's Predictions	50
46. Example MLB Dynamic Survey Open Question Response and Percentage of Other Participants Will Pick the Same Winner Question	51
47. Example MLB Dynamic Survey Final Confidence Questions	52
48. Example MLB Dynamic Survey User Feedback Page	52
49. Example MLB Dynamic Survey Thank You Page	53
50. Graphs Comparing Z-Scores of Average Money Earned to Average Number of Predictions Correct	59
51. Graphs Comparing Z-Scores of Average Money Earned to Average Amount of Error in Spread Prediction	60
52. Average User Performance (Z Scores) Compared to Outlier Score (Z Score)	63
53. Average User Performance Compared to Support Score Per Game	65
54. Kernel Densities for Support Scores	66
55. Average User Performance Compared to Time Taken Per Prediction	67
56. Average User Performance Compared to Absolute Difference in Initial Probability Prediction and Final Probability Prediction	69
57. Average User Performance Compared to Absolute Difference in User's Probability Prediction and Vegas's Probability Prediction	70
58. Kernel Density of Absolute Difference Between Each User's Initial Predicted Probability and Vegas Probability	71

59. Kernel Density of Absolute Difference Between Each User's Final Predicted Probability and Vegas Probability	72
60. Average User Performance Compared to Absolute Difference in User's Implicit Probability and Explicit Probability	73
61. Kernel Density of Absolute Difference Between Each User's Initial Implicit Predicted Probability and Explicit Predicted Probability	74
62. Kernel Density of Absolute Difference Between Each User's Final Implicit Predicted Probability and Explicit Predicted Probability	75
63. Performance of Top Predicted Performers Using Outlier and Support Score (Linear Regression Model) By Number of Users Per Week.....	81
64. Average Performance of Top Predicted Performers Using Outlier and Support Score (Linear Regression Model) By Number of Users	82
65. Performance of Top Predicted Performers Using Outlier and Support Score (K Nearest Neighbors Model) By Number of Users Per Week.....	83
66. Avg. Performance of Top Predicted Performers Using Outlier and Support Score (K Nearest Neighbors Model) By Number of Users.....	84
67. Performance of Top Predicted Performers Using Outlier and Support Score (Random Forest Model) By Number of Users Per Week.....	85
68. Average Performance of Top Predicted Performers Using Outlier and Support Score (Random Forest Model) By Number of Users.....	86
69. Performance of Top Predicted Performers Using Most Correlated Data (Linear Regression Model) By Number of Users Per Week.....	87
70. Average Performance of Top Predicted Performers Using Most Correlated Data (Linear Regression Model) By Number of Users Across the First Five Weeks	88
71. Performance of Top Predicted Performers Using Most Correlated Data (K Nearest Neighbors Model) By Number of Users Per Week.....	89
72. Average Performance of Top Predicted Performers Using Most Correlated Data (K Nearest Neighbors Model) By Number of Users Across the First Five Weeks.....	90
73. Performance of Top Predicted Performers Using Most Correlated Data (Random Forest Model) By Number of Users Per Week.....	91
74. Average Performance of Top Predicted Performers Using Most Correlated Data (Random Forest Model) By Number of Users Across the First Five Weeks.....	92

Chapter 1

INTRODUCTION

A new startup company, Unanimous AI, has created a software platform that allows individuals to create a swarm that works together to make a collective decision. Although the research is not conclusive, these human swarms have shown that they are capable of significantly increasing the average decision-making capabilities of the group and the individuals involved. These human swarms range in size from 10-50 participants and come together to answer questions on any topic. Moreover, studies conducted by Unanimous AI have shown that the size of the swarm and prior knowledge of the individuals who participate in the swarm, greatly affects the overall performance of the swarm.

In order for Unanimous AI to create the most intelligent human swarms, they need to find the most intelligent individuals to participate in each swarm. They currently chose individuals to participate in swarms by vetting individuals with surveys. These surveys are meant to measure how intelligent each individual is on the topic that the swarms is investigating. For example, if Unanimous AI is hosting a human swarm to predict the outcome of a Major League Baseball game, they would first have a group of individuals take a survey on Major League Baseball and depending on how those individuals performed, they would be hand selected to participate in the swarm. The process of individuals entering a swarm is summarized in Figure 1.

Figure 1: Unanimous AI's Process of Weeding Out Users

This process of measuring a user's knowledge on a topic before they enter a human swarm is a relatively new project for Unanimous AI, and their methodology for selecting users can be improved. Because there has not been any documented research on this topic, this study is an experimental approach on creating a new and improved methodology for choosing swarm participants. The structure of the survey, data collected, and the way that data is used to predict the knowledge of users can be improved with a new type of survey, different types of data collection, and different data analysis methods.

By improving the process used to weed out participants that go into a swarm, Unanimous AI can increase the knowledge of their individual human swarms and therefore will receive more accurate answers from the questions Unanimous AI asks their swarms.

This study begins by creating a new type of survey that is designed to better capture a user's knowledge on the topic the swarm will be asked questions about. After several iterations of developing this new survey, I performed data analysis on all the data that was collected and discovered which data was most correlated with user performance. Using the information that I gathered from data analysis, I created machine learning models that predicted the performance of each user solely based on their input to the new survey. Using these machine learning models, I ordered the users from the predicted most intelligent users to the predicted least intelligent users on the topic. From this ordering, I simulated taking the top predicted performers for each week and measured the improvement in the group's average performance as a whole.

Chapter 2

BACKGROUND

2.1 Swarm Intelligence

Swarm Intelligence is a rapidly growing sector of artificial intelligence that recreates natural and artificial systems composed of many individuals that coordinate their actions using decentralized control and self-organization in order to accomplish a task [1]. In particular, it seeks to capitalize on the superior natural design of social systems, such as flocks, to harness its power to solve human problems. Many species of unintelligent individuals amplify their group intelligence in nature by forming flocks, schools, colonies, or swarms [5]. Whether ants, birds, bees, or fish, nature repeatedly demonstrates that social creatures that work together, as unified systems, outperform the majority of individual members in decision making or problem-solving tasks [5]. For this reason, humans are looking to nature for solutions and creating biomimetic systems such as artificial swarms.

One area where Swarm Intelligence has become useful is in combining people's thoughts or predictions all in a real-time effort into a single coordinated thought or prediction. Schools of fish are able to detect subtle tremors in the water. Swarms of bees are able to communicate using high speed vibrations while they are in a swarm. Humans, contrary to other species, lack the subtle connections of other species to communicate with each other and establish tight feedback-loops [5] and must resort to other methods. For this reason, among many others, it is important to evaluate how groups of humans communicate. Group Psychology is a field of research that addresses how humans collaborate as social groups and is discussed in Subsection 2.2.

2.2 Group Psychology

Humans owe their evolutionary success to their ability to collaborate in social groups [3]. Contemporary social platforms often utilize collaboration via asynchronous polls that conclude with a majority vote as opposed to groups coming to a conclusion through discussion. For

example, decisions are often made utilizing independent votes on favored decision often as “like” or “up-votes” calculate the average vote [2]. Recent studies suggest that these asynchronous polls greatly distort group-wise decisions by introducing a biasing effect known as *herding* [4]. Herding is the process of coming to a group decision that is neither reflective of the group opinion nor optimal. It is the consequence of ineffective circulation of ideas which results in positive feedback loops.

Another more popular group decision method utilizes unstructured group discussion followed by voting [10]. In a discussion group, one member will state the problem, which is followed by an unstructured group discussion that generates information and pools judgement among participants. Typically, the meeting concludes with a majority vote based on priorities or a consensus decision [10]. Although this method aggregates ideas, it does so inefficiently, resulting in herding as well. Although not commonly used, there are other group decision making methods that try and combat the negative effects seen in asynchronous polling and group discussions including *nominal group technique* and the *delphi technique*.

In the *nominal group technique*, group members introspectively generate ideas on the provided problem or task [10]. Each member presents one idea to the group without feedback from the rest of the group. These ideas are recorded for later discussion by the group. After all of the individuals express their ideas to the group, the group deliberates. The meeting concludes with silent independent voting on priorities by individuals through a rank ordering or rating procedure [10]. The group decision is the pooled outcome of the individual votes.

Another group decision making technique is the *delphi technique*. This technique involves separating each member from the group in such a way that no two group members can collaborate. There are different ways the delphi technique is administered. Typically, each group member is given a questionnaire which contains information on the problem or the task that the group is required to work on. All of the group members contribute ideas on the given problem or task. Upon completion, all of the ideas are summarized into a report which is then returned to all

of the group members along with a second questionnaire. This organization allows individuals to probe the ideas generated by the first questionnaire.

On receiving the feedback report, each of the group members independently evaluates it and responds to the second set of questions. Upon completion of the second questionnaire, all of the members' responses are pooled together [10]; this aggregated response becomes the collective decision. By reducing the amount of social interaction, the delphi technique limits the scope of group member decisions to those with prior information provided to them and their best judgement. This lack of social interaction eliminates members from being swayed by any other member based on their thoughts about that individual member. On the other hand, it also does not allow participants to see other participant's confidence in their opinions. Therefore, the group is vulnerable to uneducated users voting on the topic.

Out of the group, nominal, and delphi decision making techniques, the delphi technique is the only one that is not affected by herding behavior. Unfortunately, the delphi technique traditionally takes the most time and is vulnerable to less educated users distorting the decision of the group. Unanimous AI's Unu platform almost mimics the delphi technique in real time, allowing users to add their input and avoiding the negative effects of herding. Lamentably, the Unu platform is still susceptible to uneducated participants negatively affecting the decision of the group. By improving the participant selection process, as shown in this study, Unanimous AI can reduce and potentially eliminate the negative effects uneducated users play in their human swarms.

2.3 Unanimous AI and Unu

Unanimous AI is an innovative research company that utilizes human swarms for multiple types of real life applications. The software platform, Unu, groups users online to collectively answer questions, make decisions, and solve problems. It imitates biological swarms and therefore enables online groups of humans to work together in real-time synchrony, collaboratively explore a decision-space, and converge on an optimal solution in a short amount

of time [2]. The result is a coordinated swarm intelligence system that produces coordinated thoughts and predictions. Unanimous AI calls this technique of gathering a collaborative group of peoples' decisions *social swarming* [2]. These online groups of people working together are commonly referred to as *swarms*.

Within this software application is a graphical puck, which represents the consensus of the group. A user's magnet coordinate reflects opinion. Each distinct possible choice is methodically spread out evenly around the exterior of the swarm. This structure allows users to freely move the magnet's coordinate to apply a force vector proportional to the distance to the coordinate. Each user's graphical magnet has a variable size, and the magnitude of the pull on the graphical puck ideally represents user *confidence*, which is a measure of how certain they are in a target. Initially, every user has the same magnet pull strength. If a user changes their prediction, the size of their magnet and magnitude of pull on the puck decreases to reflect diminished confidence. During a swarm, the puck location changes based on coordinated, dynamic feedback in a closed loop containing all of the users participating; this is not limited to any individual [2]. Consequentially, users in the swarm can synchronously coordinate the movement of the puck in real time; in other words, the whole group can come to a unanimous decision.

Figure 2: Example of a Social Swarm Being Used to Predict the Outcome of the Kentucky Derby

Unanimous AI's Unu platform has made more accurate predictions than independent participant predictions as well as predictions cumulatively averaged across a group of people. For example, a recent study conducted by researchers at Unanimous AI and Oxford University compares individual predictions to swarm predictions for fifty English Premier League (EPL) soccer games over five consecutive weeks. This study demonstrates that individuals amplified their prediction accuracy to 72% from 55% when predicting together in a swarm as opposed to as individuals [5]. In another demonstration, Unu out-predicted 20 self-identified horse-racing enthusiasts; it predicted the top four winners of the 2017 Kentucky Derby in order, a \$20 bet which returned \$11,000 [6]. The Unu platform has also correctly predicted the TIME's Person of the Year for 2016 and 2017, the outcome of Super Bowl LI, as well as out-predicted the majority of experts in the 2015, 2016, and 2017 Oscars [8]. The Unu platform is a new, innovative way of communicating people's opinions quickly and efficiently.

2.4: Current Surveys

2.4.1: Using Surveys to Weed Out Unu Participants

The more intelligent and informed the swarm population is the more informed and more accurate their predictions [9]. Therefore, in order to make smarter swarms, Unanimous AI needs to weed out poor swarm candidates using surveys. There are two methods to identify superior swarm candidates. The first way is to identify people that have historically performed well. Another way is to identify high performers based on their responses to a survey containing strategically asked questions conducted on the same week of the swarm. Each of these two techniques is used to either gather or eliminate good performers or poor performers respectively.

Historical data more easily differentiates users than their swarm answers, but it has a couple drawbacks. First, it eliminates the possibility of removing users during the first part of the sport season because there is not any historical data near the beginning of the season. Furthermore, users perform significantly worse during this period than in the rest of the season. Unanimous AI researchers believe that a lack of prior knowledge of the team dynamics; each team often has new players, coaches, formations, in addition to numerous other variables. Because the first part of the season does not reflect user performance for the rest of the season, it is discarded, further reducing the opportunity to use historical data to weed out users the first part of the season. Additionally, Unanimous AI finds is that the average performance of a user is not highly correlated week-to-week [9]. In an internal study, Unanimous AI found that their best performers are about 7% better than the average performers. Despite notably superior performance, they have weeks where they perform poorly [9].

Another internal Unanimous AI study demonstrates that the individual performance of users increases 12%, on average, when using machine learning methods in conjunction with individual survey responses (to curate the data) [9]. This study was conducted on the English Premier League (EPL) over a 12-month period. Each of the user's predictions as well as their

survey question responses were used as input to a neural network. Subsequently, poor users were eliminated.

In this study, users had to pass a test which ensures that each user is knowledgeable in the basics of EPL, which consisted of basic questions that average informed EPL fan should know such as the following,

“Who was relegated last year?”,

“Who are the reigning champions?”,

“Who is your favorite team / player?”

Qualified users were allowed to answer surveys that were posted each week to be curated by the neural network.

There is approximately an average of 10 games each week. The survey used contained multiple questions for each game, as well as a few additional useful questions. They contained three questions for each game including the following,

1. “Who will win the game?”

[Team A by 2+, Team A by 1, Tie, Team B by 1, Team B by 2+]

2. “How much would you bet on this outcome?”

[\$0-\$100 scale]

3. “What percent of people will vote the way that you did?”

[0%-100% scale]

The questions were then shown to the user in the format shown in Figure 3.

2 Chelsea @Southampton

☒ Chelsea wins by 1

☐ Chelsea wins by 2+

☐ Tie

☐ @Southampton wins by 1

☐ @Southampton wins by 2+

3 What percentage of the people taking this survey will pick the same winner you did?

0% 100%

18 Clear

4 You can bet \$0 to \$100 on this outcome, regardless of goal margin. How much would you bet?

\$0 \$100

14 Clear

Figure 3: Example Questions for a EPL Game Prediction

The latter question was added in later in the data collection process, therefore the study focuses only on the first two questions.

Each subject was also asked a series of questions that included all of the games for that week, see Figure 4. First, they were asked their pick of the week (the team that is most likely to win). They were then asked to place a bet, \$0 to \$100, on that pick of the week to assess confidence.

32 Pick of the Week: Of all your picks, which do you think is most likely to be correct?

☒ Chelsea @Southampton – Chelsea wins	☐ Bournemouth @Liverpool – Bournemouth wins
☐ Chelsea @Southampton – Tie	☐ Bournemouth @Liverpool – Tie
☐ Chelsea @Southampton – @Southampton wins	☐ Bournemouth @Liverpool – @Liverpool wins
☐ Leicester City @Burnley FC – Leicester City wins	☐ Man. City @Tottenham – Man. City wins
☐ Leicester City @Burnley FC – Tie	☐ Man. City @Tottenham – Tie
☐ Leicester City @Burnley FC – @Burnley FC wins	☐ Man. City @Tottenham – @Tottenham wins
☐ Brighton @Crystal Palace – Brighton wins	☐ Arsenal @Newcastle – Arsenal wins
☐ Brighton @Crystal Palace – Tie	☐ Arsenal @Newcastle – Tie
☐ Brighton @Crystal Palace – @Crystal Palace wins	☐ Arsenal @Newcastle – @Newcastle wins
☐ Watford @Huddersfield – Watford wins	☐ West Brom @Man United – West Brom wins
☐ Watford @Huddersfield – Tie	☐ West Brom @Man United – Tie
☐ Watford @Huddersfield – @Huddersfield wins	☐ West Brom @Man United – @Man United wins
☐ Everton @Swansea City – Everton wins	☐ Stoke City @West Ham – Stoke City wins
☐ Everton @Swansea City – Tie	☐ Stoke City @West Ham – Tie
☐ Everton @Swansea City – @Swansea City wins	☐ Stoke City @West Ham – @West Ham wins

33 You can bet \$0 to \$100 on this outcome. How much would you bet?

\$0 \$100 [Clear](#)

34 What percent of the people who take this survey will get their Pick of the Week correct?

0% 100% [Clear](#)

35 Tie of the Week: Which match do you think is most likely to end in a Tie?

☒ Chelsea @Southampton	☐ Bournemouth @Liverpool
☐ Leicester City @Burnley FC	☐ Man. City @Tottenham
☐ Brighton @Crystal Palace	☐ Arsenal @Newcastle
☐ Watford @Huddersfield	☐ West Brom @Man United
☐ Everton @Swansea City	☐ Stoke City @West Ham

[Prev](#) [Next](#)

Figure 4: Example Questions for All EPL Games On a Given Week

Next, they were inquired as to their overall support from other users to their prediction. Then they were asked which game they thought was most likely to end in a tie.

Finally, each user was asked four questions to obtain metadata on the users. They include the following,

1. What is your level of knowledge about the teams playing this weekend?

[1-No knowledge, 2, 3, 4, 5-Expert]

2. What is your level of confidence about the picks you just made?

[1-No knowledge, 2, 3, 4, 5-Expert]

3. What percent of games you picked do you think you will get right?

[0% - 100%]

4. What percent of games do you think the average person will get right?

[0% - 100%]

The way they were asked to the user is shown in Figure 5.

* 36 What is your level of knowledge of teams playing this weekend?

1 - No knowledge at all 2 3 4 5 - Expert

* 37 What is your level of confidence in the picks you just made?

1 - Not confident at all 2 3 4 5 - Extremely confident

* 38 What percent of the games you picked today will you get right?

0% 100% 61 Clear

* 39 What percent of the games will the average person get right?

0% 100% 53 Clear

When you click DONE, your picks will be recorded. Please refer to the HIT for instructions to join the swarm.

◀ Prev Done ▶

Figure 5: Example Metadata Questions for the EPL Survey

The first two questions measure how much the user thinks they know while the last two were meant to measure their metacognition, or how much they know about how much they know regarding the EPL. After each week, each user's game prediction performance was calculated relative to the performance of all of the other users by comparing the z-score; 0 represented average, whereas positive and negative represented good and bad performers, respectively.

Three of the measurements proved to be significant in predicting a user's performance. The first measurement, *outlier score*, is a calculation of the percent of the time each user voted

against the crowd for a week. This measurement seems highly indicative of true knowledge of the outcome of a match, or at least knowledge about the match. An outlier score above 50 % indicates that the user predicted against the majority half of the time and therefore was likely randomly guessing on a lot of the games. Conversely, an outlier score of 10% or less indicates that the user is generally knowledgeable, however might occasionally go against the majority (and be more trustworthy when they do). The second measurement is called the *support calculation*. The support calculation was found by normalizing the number of users that voted with them for each question and all of the games for that week. For example, 70% crowd agreement adds + 0.7 added to a user's total for the week. The support calculation is essentially a culmination of the outlier score for all the games that week.

The third significant measure was the users' *return on investment (ROI)*. For each game the users predicted their expected winner and gave a dollar amount (virtual money) from \$0 to \$100 that they were willing to bet on their expected winner. After the user completed the survey and the sporting event concluded, the users ROI was calculated. The user earned an amount of virtual money equal to that they bet on every correct prediction and lost that amount of money on each prediction that they got wrong. From this number, the users bet portrays their confidence. Ideally, users bet a lot of money on the games they were confident about and bet less on games they were not confident about. If a user is good at this they would receive a higher ROI than a user who is not. Consequentially, they are better at portraying their confidence than other users.

This study demonstrates that there is a lot of value in a user's prediction confidence because it indicates whether a user is aware what they know. Furthermore, the amount that a user bet on each prediction against the support for that prediction is a superior metric. From these two metrics alone, it is clear that a good user places large amounts of virtual money on games that are heavily supported by other users (essentially, strongly favored teams) and a lot less on games where there was less support. Thus, this study demonstrates that the best results come from translating these two features as well as meta data into a neural network to predict the best potential candidates for an informed swarm.

The performance of the group varied with the size of the group, where the optimal size was found to be the top 70% best predictors from the total. By choosing the top 70% of the users, it increased the average individual performance from about 50% correct up to 62% correct. This 12% increase is substantially greater than the 7% increase seen by weeding users out utilizing their historical data.

2.4.2 Using Surveys to Create Single Collective Predictions

Surveys like this can potentially be aggregated to make single predictions. Collecting each user's individual predictions, it is possible to calculate the average prediction or the most popular prediction across all of the user's individual predictions. This single culminated prediction is typically more accurate than any single user. Although the average prediction calculated across a group of predictions is more accurate than any individual prediction, data suggests that swarms predictions are more accurate than the collective average.

Figure 6: Example of a Swarm Being Used to Predict the Outcome of a MLB Game Between the Chicago White Sox and the Boston Red Sox

Unanimous AI's results suggest that polls accurately capture the average views of a population, but without real-time feedback, they do not allow groups to explore different options and find consensus synchronously [2]. Researchers at Unanimous AI believe that the reason that swarms are much more successful because they allow users to update their opinion in real-time while assessing combining the information contained by users [2]. Throughout this process, each user continuously assesses their opinion and compares it to the other user's convictions. This forces the user to weigh their confidence and preference in their opinion in real-time. With all the users doing this in unison, the swarm is able to collectively converge to one outcome that maximizes the overall confidence and cohesion within the group [2]. The researches at Unanimous AI believe that the swarms' real-time feedback and collaboration is able to efficiently capture the group's wisdom, therefore, improve predictions.

Figure 7: Thought Process While a User Is In a Social Swarm

Unfortunately, the only way to way to currently get collaborative feedback is to place individual users into a swarm and allow them to receive continuous feedback. This is not replicated in a survey; instead a user is without input from any other source. Without the input of data that occurs during a swarm, thoughts and confidence are not exchanged and therefore there is less information being aggregated in the making their prediction.

Figure 8: Thought Process While a User Takes a Survey

Unanimous AI believes there are three advantages that swarms have over current surveys which includes the following,

1. Swarms allow users to consider all the alternative outcomes.
2. Swarms allow the users to see all other user's choices in real time.
3. Swarms allow the users to express their confidence in an answer to the rest of the group, as well as see every other user's confidence in their answer.

Researchers at Unanimous AI hypothesize that utilizing this information will increase survey accuracy.

SYSTEM DESIGN

3.1 Initial Concept

3.1.1 Dynamic Survey

The original goal of this project was to incorporate benefits of a swarm into a survey, by evoking and collecting behavioral information and ultimately running machine learning on that behavioral data. It was theorized this implementation would generate more accurate predictions than a standard survey and better measure users' confidence. In order to mimic a swarm interaction, ideally the surveys would provide every other users' opinions. One of the benefits of surveys is that they allow the users to independently submit information during their free time. Unfortunately, it is difficult to effectively provide every user every other user's input. Furthermore, each user is preferentially receiving the same information from every other user so that the survey does not contain any data inconsistencies amongst users.

In order to maintain consistency, but also provide another opinion and mimic a swarm like input, a second opinion or *expert opinion*, or knowledgeable authoritarian opinion, should be provided to a user after they input their initial prediction. The initial survey design had the following user interaction and layout which is subsequently explained.

- 1) A user gets prompted with a question and a slider.**
- 2) The survey would keep track of the user's result, as well as track the speed and dynamics of how the user moved the slider.**
- 3) When the user submits this initial prediction, the survey would give a perturbation stimulus to the user.**
- 4) After the perturbation is shown, the user is then given the opportunity to update their prediction.**

First the user gets prompted with a question slider. An example is given in Figure 9, below.

Who will win and by how much, the **San Francisco 49ers** or the **Oakland Raiders**?

Figure 9: Example of an Initial Dynamic Survey Question and Slider

We wanted to control the exact timing that the user read the question and provided a response.

Secondly, the researchers at Unanimous AI desired to reflect the confidence in user expression using speed and cursor dynamics.

The most important commodity of this new survey is that it can be used to track behavioral information of the user, not just the final resting position of the slider. This type of behavioral information included the time delay (the difference between when the user is prompted to move the slider and when they actually move it), the speed of the slider, when it is being moved, and the motion dynamics of the slider. An example for a question, given below, is illustrated in Figure 10.

Who will win and by how much, the **San Francisco 49ers** or the **Oakland Raiders**?

Figure 10: Example of an User's Initial Estimate to the Initial Dynamic Survey Question and Slider Model

As an example, at the end of this process, the interface might be positioned by the user to 49ers by 9 points.

Next, the user submits the initial prediction and gets a perturbation, in Step 3). This could be an indicator describing the proclaimed expert prediction as well as the machine learning classifiers prediction and the crowd's prediction. The primary concern is psychological. We want this perturbation to inform users of authoritarian, alternative views. This perturbation could potential be fictitious (although the user would believe it is real) or this perturbation could be real (based on expert, artificial intelligence (AI), or crowd data). The perturbation could potentially be randomly selected and could disagree with the user or could imitate an actual expert opinion. This perturbation could be presented, for example, as follows in Figure 11.

Who will win and by how much, the **San Francisco 49ers** or the **Oakland Raiders**?

Figure 11: Example of an User's Initial Estimate and Application Perturbation on the Initial Dynamic Survey Question and Slider Model

In Step 4), the user is given the opportunity to update their prediction. Once the user completes their prediction they indicate that they are done by clicking a button. A timer could also appear, forcibly making them decide quicker, in a psychologically pressuring environment. The

software again tracks how long it takes the user to answer this question, as well as the time delay between the prompt and the user grabbing the slider, the speed of the slider motion, the trajectory of the slider motion, and speed of user update to the new answer. Conclusively, the goal was to use this additional behavioral data, in response to the perturbation stimulus, as a further indicator of confidence and conviction in the answer given, see Figure 12.

Figure 12: Example of an User's Final Estimate After the Application's Perturbation on the Initial Dynamic Survey Question and Slider Model

One of these surveys will theoretically encompass a set of questions (for example all 15 NFL football games being played during a given week) and a large population (for example, 100 football fans). From all of this data, a more detailed and expressive data set contains information that indicates final prediction across all users as well as behavior.

3.1.2 Machine Learning

Traditional *wisdom of the crowd* for sports predictions is gathered from individual user predictions in surveys and calculated as the averages of those predictions. This method is often used to predict different outcomes such as the winner of a game or final score of a game. For example, if there were 100 users predicting the 49ers / Raiders game, the process would generate an average value across the data set producing a mean. That final result would likely be slightly more accurate than the vast majority of individuals on their own.

Unfortunately, there are several problems with just taking the average answer. First, every user in a group has a different level of confidence and conviction in their answer although they are equally weighted. Second, each user poorly expresses or even interprets the magnitude of their confidence. Third, if asked to report their confidence, every individual has a very different

internal scale, so confidence cannot be accurately averaged. Therefore, traditional methods fail because they average predictions of users equally weighted, but each user does not have equal confidence models.

Given this information, Unanimous AI wanted to utilize behavioral data to optimize the aggregation of forecasts across a population of users, to most efficiently weigh the relative significance of the predictions made by the users across the population. Unanimous AI hoped to test whether this process can be employed to create wiser crowds of users being surveyed. Unanimous AI found that it did.

3.2 Dynamic Survey Requirements

I needed to create a system with the model to improve the survey. First, I needed to create design requirements. My initial development prototype had the following requirements:

1. Easy to deploy to users
2. Self-explanatory user interface
 - a. Each of the user's tasks were easy to understand
 - b. Each of the graphical sliders were easy to use
3. Able to store each of the user's initial inputs, as well as how long it took the user to answer each question
4. Able to show the users another opinion in a clear manner for each question
5. Able to store the user's second inputs, as well as how long it took the user to answer each question.

With these requirements established, the system was ready to be designed and implemented.

SYSTEM IMPLEMENTATION

4.1 Dynamic Survey First Prototype

Unanimous AI requires that the program is an easily deployable and accessible web application, therefore I created a web application. Pre-existing web frameworks did not fit the requirements. Instead, the application was developed using HTML, CSS, and JavaScript. Google Firebase was used to host the website and as its database because it is open source, easy to use, and quick to deploy. A link to the program is provided to each user that is easily accessible from any web browser. Each user answers the survey using their web browser and their data is stored into Google Firebase's non-relational database.

Figure 13: Diagram of the Data Flow of the Web Application

Bugs were eliminated from the first prototype by testing on family and friends, predicting NBA matches. Next, the survey was sent to 33 individual NBA fans who were each paid \$2 to take the survey. All of the survey participants were required to successfully complete preliminary quiz to ensure that they had basic knowledge of the NBA before they could participate. Users were chosen from Amazon's Mechanical Turk service, which allows people to get paid for

different tasks (there is a large number of eager sports fan who are willing to offer their predictions for a small amount of money). The survey was presented as follows,

1. Mechanical Turk users were instructed to enter in their Mechanical Turk Worker ID at the website. A JSON object was subsequently created in the database to keep track of all the user's inputs.

Figure 14: Screenshot of the Initial Login Screen

2. Once an individual user had logged in, they were sent to the instructions screen.

Figure 15: Screenshot of the Instructions Screen

The instruction screen included information to further motivate the users with monetary rewards. The survey of NBA predictions also included two competitions. The top three winners of each competition would receive an additional cash bonus. The top three winners of the first competition were the three users with the smallest differences between their point spread prediction for each game and the actual score of each game. The three winners of the second competition were the three users that won the most virtual money by betting correctly on perceived winner each game. For each competition, first place would receive an additional \$25, second place would receive an additional \$10, and third place would receive an additional \$5. Unanimous AI funded these payments.

3. After the user had understood the instructions, they were asked to make predictions on each of the NBA matches that week. They were first asked to make a prediction on the final score of the game before any external stimulus.

Figure 16: Screenshot of an example NBA First Prediction Screen a user would have seen

4. Subsequently the user was shown the *expert prediction* and was given the opportunity to make a new prediction given this perturbation. The expert prediction used was the betting point spread that Vegas was using.

Figure 17: Example of a User's Second Prediction Given an "Expert's Prediction"

The whole point of the expert's prediction was to present another point of view that would normally be present in a swarm. This additional perturbation made the user reflect on their initial thought and confidence, hopefully, recreating a similar thought process that a user would normally go through in a swarm as seen in Figure 18.

Figure 18: User’s Thought Process Once Exposed to the “Expert’s Prediction”

Initially Unanimous AI wanted to pressure users with a timer while they made predictions. After discussion, Unanimous AI engineers and I decided that one of the advantages of a survey is that there is no time limit which allows the user to do more research with every question if they need to. Having the ability to conduct more research was decidedly more valuable than trying to mimic the pressure felt by a user in a swarm.

5. Once the user made their second prediction, they completed a probabilistic wager slider, which replaced the dollar confidence slider that was in the current survey (shown in Figure 19).

Figure 19: Screenshot of an Example Probabilistic Wager Slider for a Given Game

The point of the proposed dollar confidence slider is to measure a user's confidence in their pick. However, individual users have different ways of depicting confidence. Some users are risk tolerant and normally bet \$ 80, while others are risk-averse and average \$20 per a bet. Users judge confidence differently and therefore it is hard to compare universal confidence and almost impossible to determine how confidence translated across users. Thus, a different slider was proposed to measure the user's confidence.

This advanced probabilistic slider came from one of the engineers at Unanimous AI, Gregg Wilcox. It is based on a Brier score for each team. The user moves the slider left or the right and wins the amount of money corresponding to the option of the team that actually won. For example, if the user thought the Pistons would win, they would place more virtual money on the Pistons even if that meant placing less money on the Raptors if they won. If the user believed there was an equal chance of either team winning, they would keep their wager in the middle, which placed a wager of \$75 for either team. If the user was 100% confident that the Pistons would win they could move the slider all the way to the right and would win \$100 if the Pistons won and \$0 if the Raptors won. This new slider is thought to better indicate the user's confidence because it forces the user to place a probability on a team winning, as well as place a confidence on that probability. Furthermore, this new slider gamified the task so that the user was motivated to give their best prediction to win kudos.

This new probabilistic slider uses the Brier Score, which verifies the accuracy of a probability forecast by minimizing loss [12]. The Brier Score is most generically calculated using the mean squared error function on binary outcomes, such as team 1 winning or team 2 winning. In the typical calculation of the Brier Score the function in Eq. (1) is used [12].

$$BS = \frac{1}{N} \sum_{t=1}^N (f_t - o_t)^2 \quad (1)$$

N = the number of items you're calculating a Brier Score for

f_t = forecast probability

o_t =outcome (1 if it happened, 0 if it didn't)

The proposed method calculates one cost at a time and the output is converted into a dollar format so that the slider was gamified for the user. Therefore, the Brier Score I used looked like the function seen in Eq. (2), where each dollar amount below each team represented the outcome of that team winning.

$$\text{Brier Score} = 100 * (1 - (f - o)^2) \quad (2)$$

f = user forecasted probability of one team winning

o = possible outcome of the match ($o = 1$ if that team won, $o = 0$ if that team lost)

For example, if a user was 80 % confident that one team would win, they would drag the thumb of their slider closer to that team. Below the team that the user predicted to win, \$96 would be observed (using values $f = 0.8$ and $o = 1$). \$ 36 would be observed under the other team (using $f = 0.8$ and $o = 0$).

Once the user had completed the probabilistic wager slider for a prediction, they would either predict the next game, if there were more games that week, or they would complete the survey. If there were more games to predict that week the user would continue to answer questions identical to steps 3 through 5 for different teams, otherwise they would be given a summary of their predictions. Continuing on with the survey,

Thank You For Your Predictions!				
1) Here is a summary of your predictions!				
Matchup	Your Predicted Winner	Your Predicted Spread	Your Predicted Probability	Virtual Money Distribution
Spurs vs Warriors	Warriors	+4	63%	Spurs: \$62.79 Warriors: \$84.79
Raptors vs Pistons	Raptors	+5	64%	Raptors: \$59.04 Pistons: \$87.04
Heat vs 76ers	76ers	+4	60%	Heat: \$65.19 76ers: \$83.19
Pelicans vs Trail Blazers	Trail Blazers	+6	65%	Pelicans: \$59.04 Trail Blazers: \$87.04
Bucks vs Celtics	Celtics	+4	57%	Bucks: \$64.00 Celtics: \$84.00
Pacers vs Cavaliers	Cavaliers	+6	65%	Pacers: \$66.36 Cavaliers: \$82.36
Jazz vs Thunder	Thunder	+4	55%	Jazz: \$68.64 Thunder: \$80.64
Timberwolves vs Rockets	Rockets	+5	63%	Timberwolves: \$66.36 Rockets: \$82.36

Figure 20: Screenshot of an Example Summary of Predictions Table

7. Once the user was given the opportunity to review their predictions, they were asked which team they thought was most likely to win and why.

2) Out of all the predictions you have just made, which team do you think is the most likely to win and why? Try to be specific in your reasoning.

Enter text here...

Figure 21: Screenshot of a Favored Team Questionnaire

The reasoning behind the open answer questionnaire was to potentially run machine learning on user inputs to see if there were any trends in what good predictors input into this section versus bad predictors. Although running machine learning on user's open-ended question was not implemented in this study, the data was still collected and stored in case of a future study.

8. Once each user input their favored team for the week, they were asked if there was anything they would add to improve the survey.

The screenshot shows a web browser window with the address bar displaying "Secure https://unanimousipoll.firebaseio.com". The page has a teal background and contains the following text:

Optional: Answer the Questions Below

Is there anything you think we should change about the entire questionnaire that you just completed? What was good? What was bad? Were there any parts that were confusing? Any input would be helpful!

Below the text is a large, empty white rectangular box for user input. At the bottom of this box is a grey button labeled "Submit".

Figure 22: Screenshot of a Favored Team Questionnaire

A summary of the user's steps through this survey can be seen in Figure 23.

Figure 23: Summary of User's Navigation Through the Initial Prototype

The data that was collected was stored in a JSON format, where each user's data was stored based on the sport, date, and each user's Mechanical Turk Worker ID. Each of the user's predictions for individual matches were stored based on what two teams were playing in the game they predicted on. The user's answers to the open-ended questions were stored internal to the user but externally to any of the match data. Each user just simply filled out the survey, but the more complex way the data was stored can be seen in Figure 24.

Figure 24: Example Data Saved From a User Predicting 8 Games One Week

Figure 24 exemplifies data storage structure for each user. As shown in Figure 24, a new JSON object was created for every user based on the sport and the date. Within that new object, an internal object was created for each game the user made predictions on. Every other question outside of the individual game predictions was stored internally to the user object.

4.2 Dynamic Survey First Prototype Feedback

Users prefer this new layout to the old one according to the feedback received from users in the Step 8 of Figure 23 and below,

Is there anything you think we should change about the survey?

Although there was lot of positive feedback, there was still some necessary changes for Iteration 2. First, users were exposed to the expert prediction before they input their dollar confidence prediction in the money wager slider. Therefore, it could bias their predictions, which is not desirable. Thus, it was decided that the dollar confidence slider should be put on the same page as the spread slider in order to ensure that no initial predictions were corrupted by the expert opinion. Moreover, the probability slider was also included on the same page in order to make the comparison of how users interacted with this new money slider (that used the Brier score when compared to a 0 to 100 probability slider) better. Having all of these sliders on the same page enabled the database to store all of the user's initial predictions before an expert's prediction for each slider, as well as all of their predictions after the perturbation.

Finally, because the majority of the subsequent data collected would be on the MLB and not the NBA, the expert prediction perturbation was moved to the probability slider instead of the spread slider. This was because the point spread of the MLB is a lot smaller than the NBA. Therefore, Vegas places the predicted point spread at +1 on the favored team almost every time, which would not provide much stimulus for the users taking the survey.

4.3 Static Survey Overview

The Mechanical Turk users took two surveys. The users took the dynamic survey prototype and the old static survey. This is done to compare and contrast the results. The static survey gathered similar information and was laid out as follows:

1. First the individual users signed in using their Mechanical Turk Worker Identification.

Thanks for being part of this week's NBA swarm.

There are two parts to participating in Unanimous A.I.'s NBA swarm:

First, you will record your personal picks for tonight's NBA games in this survey. **Complete this survey before 1pm ET today.**

Then, you and a group of NBA fans will join together in UNU and make your picks as a swarm intelligence **at 1pm ET today**

This short survey enables us to compare individual members of our swarms to how the swarm performs as a collective intelligence.

STARTING THIS WEEK, and in the NBA only we are experimenting with a new survey pilot. We are paying \$25, \$10, and \$5 bonuses each in TWO categories. We'll pay that amount to the persons who earn the most hypothetical "dollars" in the wagering game part of this as usual.

We'll also pay separate bonuses of \$25, \$10 and \$5 to the top three participants who come closest on the **point spreads** in the games this week.

All bonuses will be awarded based on responses to the other survey. We are continuing this survey in addition so that we have both data sets to compare. That's why we have added an extra \$1 to the HIT payment this week. Thanks for your help in contributing to our ongoing science!

* 1 What is your Worker ID?

Next →

Figure 25: Example Static Survey Sign In Screen

2. After each user signed in, they were given the instructions for the survey.

The scoring for bonuses for NBA this week will use our new pilot method in the other survey set.

We'll pay bonuses based on how you score with your hypothetical, imaginary money wagers in the questions on the next page.

(Again, please do not set every wager at \$100. We'll disqualify your entry and eventually simply remove your qualification.)

We'll also pay another set of bonuses based on how accurate your point spread is.

We thank you for helping us by completing both surveys today!

◀ Prev Next ▶

Figure 26: Example Static Survey Instructions Screen

3. Once each user was given the opportunity to read the survey instructions, they were able to fill out the survey, which primarily consisted of predicting all of the NBA matches for that week. The user was asked 4 questions for each game:

1.) Who will win the game? (Conference rank and record provided)

[Team 1(high confidence), Team 1(low confidence),

Team 2(high confidence), Team 2(low confidence)]

2.) How many points is your pick likely to win by?

[0-16+]

3.) What percentage of people taking this survey will pick the same winner you did?

[0% - 100%]

4.) How much would you bet that the team you selected will win?

[\$0 - \$100]

The questions for each game were shown to the user in the format shown in Figure 27.

* 2 Who will win: San Antonio (6th, 6-4) @Washington (6th, 4-6)?

☒ San Antonio wins (high confidence)

☐ San Antonio wins (low confidence)

☐ @Washington wins (high confidence)

☐ @Washington wins (low confidence)

* 3 How many points is your pick likely to win by?

0 16+ 8 Clear

* 4 What percentage of the people taking this survey will pick the same winner you did?

0% 100% 61 Clear

* 5 You can bet \$0 to \$100 that the team you selected will win (regardless of margin). How much would you bet?

0 100 63 Clear

Figure 27: Example Static Survey Questions for Predicting an NBA Game

After the user had predicted each game for that week, they were asked three questions that related to all the games that were occurring that week:

1.) Out of all the teams playing this week, which do you think is most likely to win?

[Option to choose any team playing that week]

2.) What percentage of people taking this survey will get their Pick of the Night correct?

[0% - 100%]

3.) How much would you bet that your Pick of the Night wins?

[\$0 - \$100]

These three questions given to the user as illustrated in Figure 28.

* 34 Pick of the Night: Of all your picks, which do you think is most likely to be correct?

☐ San Antonio at Washington – San Antonio wins	☐ Cleveland at Miami – Cleveland wins
☐ San Antonio at Washington – @Washington wins	☐ Cleveland at Miami – @Miami wins
☐ Denver at Toronto – Denver wins	☒ Dallas at Sacramento – Dallas wins
☐ Denver at Toronto – @Toronto wins	☐ Dallas at Sacramento – @Sacramento wins
☐ Portland at New Orleans – Portland wins	☐ Milwaukee at LA Clippers – Milwaukee wins
☐ Portland at New Orleans – @New Orleans wins	☐ Milwaukee at LA Clippers – @LA Clippers wins
☐ Chicago at Houston – Chicago wins	☐ Indiana at Golden State – Indiana wins
☐ Chicago at Houston – @Houston wins	☐ Indiana at Golden State – @Golden State wins

* 35 What percent of the people who take this survey will get their Pick of the Night correct?

0 100 50 Clear

* 36 You can bet \$0 to \$100 that your pick of the night will win (regardless of confidence). How much would you bet?

0 100 44 Clear

◀ Prev Next ▶

Figure 28: Example Static Survey Questions for Users Predictions of the Week Out of Every Game

3. Once each user input a prediction for each game that week as well as their “Pick of the Night”, they gave some metadata about themselves. Each user was asked to answer four more questions before they submitted the survey:

1.) What is your level of knowledge of the teams playing this week?

[1 (No knowledge at all), 2, 3, 4, 5 (Expert)]

2.) What is your level of confidence in the picks you just made?

[1 (Not confident at all), 2, 3, 4, 5 (Extremely confident)]

3.) What percent of the games you picked today will you get right?

[0% - 100%]

4.) What percent of the games will the average person get right?

[0% - 100%]

These questions are demonstrated in Figure 29.

Figure 29 displays four example static survey questions for users' predictions of the week out of every game. The questions are numbered 37 through 40.

- Question 37:** "What is your level of knowledge of the teams playing tonight?" The response scale is 1 (No knowledge at all), 2, 3, 4, 5 (Expert). The selected answer is 4.
- Question 38:** "What is your level of confidence in the picks you just made?" The response scale is 1 (Not confident at all), 2, 3, 4, 5 (Extremely confident). The selected answer is 4.
- Question 39:** "What percent of the games you picked today will you get right?" The response is a percentage slider from 0% to 100%. The selected value is 60%.
- Question 40:** "What percent of the games will the average person get right?" The response is a percentage slider from 0% to 100%. The selected value is 49%.

Figure 29: Example Static Survey Questions for Users Predictions of the Week Out of Every Game

4.4 First Dynamic Survey Prototype Findings

Once all users completed both surveys, the data was analyzed by a Unanimous AI engineer, Gregg Wilcox, to verify that the Dynamic Survey improved performance using curation from current survey. Because Unanimous AI wanted to move to the Dynamic Survey as the primary survey full time as soon as possible, the data analysis was conducted concurrently with the development of the second Dynamic Survey prototype. This data analysis study was only done on one week of data and therefore was not conclusive, however the study still suggested that the new Brier Score probabilistic wager method better measured a user's confidence in a particular team than the previous dollar confidence slider method. Furthermore, the study showed that 69% of users changed their point spread prediction based on the expert pick at least once, which means that allowing users to change their prediction after a perturbation round leads the accumulation of some additional useful information on how to improve users' prediction model.

One comparison between the new dynamic survey and the old static survey that proved to be useful was between the old dollar confidence slider and the new probabilistic money slider. The major issue that was occurring in the previous dollar confidence slider was that users were incentivized to place the maximum confidence on every prediction, because they would either win that money or they wouldn't. Obviously, this led to Unanimous AI receiving non informative data on a user's confidence in predictions. Although this data analysis was only performed on one week of data and was not conclusive, the data alluded to the fact that the new probabilistic money slider was a better measure of a user's confidence. The primary way the data indicated this was that the new probabilistic money slider had a stronger correlation to the user's predicted point spread, therefore was a better measurement of how confident the user was in their prediction and was a better universal scale for measuring different user's confidence.

4.5 Second Dynamic Survey Prototype

The results of initial data analysis of the first prototype show that this new method has the capability to capture the same information as the previous survey as well as how user's react

after a perturbation. Furthermore, the new probabilistic money slider solved the issue of users selecting 100% confidence every time. Therefore, after gathering all the users' feedback from the first prototype and brainstorming with multiple Unanimous AI engineers a new iterations of the Dynamic Survey was created to again be deployed synonymously with another static survey. Because the 2018 NBA regular season was coming to an end during the time of this project, the second prototype collected data on user predictions of weekly games in the 2018 MLB (Major League Baseball) season.

As shown in Figure 30 and Figure 31, the login and instructions page did not change from the first to the second prototype except that the instructions were modified to better suit the MLB instead of the NBA.

The image shows a user sign-in page with a teal background. At the top, the text "Sign In To Make Predictions" is centered in white. Below it, a line of text reads "Enter your Mechanical Turk username below and select Login when finished." In the center, there is a white rectangular input field containing the placeholder text "Enter Mechanical Turk Worker ID". Directly beneath this input field is a light gray rectangular button with the text "Login" centered on it.

Figure 30: User Sign In Page

Instructions

If you make correct predictions you can win a \$50 bonus for these predictions!
Please read the following instructions before moving forward.

- 1) Your job is to predict the outcome of 15 upcoming MLB games.
- 2) You are expected to do research on each game before you make each prediction.
- 3) There are two ways to win, by predicting the correct Point Spread or by Game Wagering. First place in each competition wins a bonus of \$25, second place wins \$10, and third wins \$5. You can win both competitions for a total of \$50!

COMPETITION 1: POINT SPREAD

The three users with the smallest differences between their Point Spread predictions and the outcomes of this week's games win!

Example: If you predict the San Francisco Giants will win by 3 and they win by 2, your difference would be 1.

COMPETITION 2: GAME WAGERING

The three users that win the most virtual money by wagering on the matches this week win!

Example (see image below): If you wager \$96.00 for Kansas City to win and \$36.00 for Cleveland to win, you will win \$96.00 if Kansas City wins or \$36.00 if Cleveland wins.

Figure 31: User Instructions Page

The layout of all the sliders as well as the addition of the probability slider. we the major changes to the layout. As shown in Figure 32 and Figure 33, the new probability slider was placed under the spread slider and above the newly relocated money slider. The probability slider imitates the dollar confidence slider from the static survey since both surveys asked the user to state their predicted probability of each team winning on a scale from 0 to 100. As stated before, the money slider is now placed on the same page as the spread slider.

What Will Be the Outcome of the Mariners vs the White Sox?

Select by how much you think a team will win.

Mariners +3 +1 +1 +3 White Sox

Select the probability that each team will win.

50% 50%

Using the distribution below, how much virtual money are you willing to wager that each team will win?
Note: You will win the amount below the team that wins.

\$75 \$75

Submit

Progress (0 of 15 predictions completed)

Figure 32: Second Prototype Slider Screen Before a User's Prediction

What Will Be the Outcome of the Mariners vs the White Sox?

Select by how much you think a team will win.

Mariners +3 +1 +1 +3 White Sox

Select the probability that each team will win.

36% 64%

Using the distribution below, how much virtual money are you willing to wager that each team will win?
Note: You will win the amount below the team that wins.

\$62.79 \$84.79

Submit

Progress (0 of 15 predictions completed)

Figure 33: Second Prototype Slider Screen With a User's Prediction

All three sliders were placed on the same page which allowed me to gather each user's initial predictions before the *crowd's prediction*, or the aggregated decision perturbation, was shown to the user. Next, the crowd's predictions perturbation was shown once each user made

their initial prediction for a particular game and selected the submit button. As shown in Figure 34, once the perturbation was shown to the user, the user was able to change their spread, probability, and money slider predictions.

Now Given the Crowd's Prediction (in Blue), What Do You Predict Will Be the Outcome? The Crowd's Prediction is the Average Forecast Across 100 Participants.

Select by how much you think a team will win.

Mariners +3 +1 +1 +3 White Sox

Select the probability that each team will win.

57% 43%

Using the distribution below, how much virtual money are you willing to wager that each team will win?
Note: You will win the amount below the team that wins.

\$79.75 \$69.75

Submit

Progress (0 of 15 predictions completed)

Figure 34: Second Prototype Slider Screen After Crowd Prediction Perturbation

In the first prototype, a lot of the user feedback pertained to confusion revolving around the expert's prediction terminology. Therefore, in this prototype the perturbation wording was changed from an *expert's prediction* to *crowd's prediction* and the user was told that the latter was the "average forecast across 100 participants," as shown in the title of the web page seen in Figure 34. This change elucidated the confusion surrounding this terminology. Since this new dynamic survey prototype still did not encompass all of the information being gathered in the static survey, each user was also required to fill out the static survey as well.

4.6 MLB Static Survey

The static survey for the MLB was similar to the static survey for the NBA. Each user was given a set of instructions (Figure 35) and asked to sign in (Figure 36) before they entered the survey. In this static survey, Unanimous AI engineers decided to collect metadata on each user in hopes that this data might be used in the future to further help predict user's performance.

Thanks for being part of this week's MLB swarm.

There are three parts to participating in Unanimous AI's MLB swarm:

First, you will record your personal picks for tonight's MLB games in this survey. Complete this survey **before 1pm ET today**.

THEN, we are experimenting with a new survey pilot. **We are paying \$25, \$10, and \$5 bonuses each in TWO categories.** We'll pay that amount to the persons who earn the **most hypothetical "dollars" in the wagering game part of this as usual.**

We'll also pay separate bonuses of \$25, \$10 and \$5 to the top three participants who come closest on the point spreads in the games this week.

The link for this second survey is in the HIT.

Third, you and a group of MLB fans will join together and make your picks as a swarm intelligence **at 1pm ET today.**

[\(What's a "swarm"? What is "Swarm Intelligence?" Here's a quick intro! \(1 minute video\)\)](#)

This short survey enables us to compare individual members of our swarms to how the swarm performs as a collective intelligence.

All bonuses will be awarded based on responses to the other survey. We are continuing this survey in addition so that we have both data sets to compare. **That's why we have added an extra \$1 to the HIT payment this week.** Thanks for your help in contributing to our ongoing

Figure 35: User Instructions for MLB Static Survey

As shown in Figure 36, the user had to provide metadata which was extracted using four questions. This was done before entering the survey and providing their Worker ID. The metadata extraction consisted of four questions presented below:

- 1.) In what year were you born?**
- 2.) How do you identify yourself?**
- 3.) What state do you reside in?**
- 4.) What is the highest level of school you have completed or the highest degree you have received?**

* ① What is your Worker ID?

* ② In what year were you born? (enter 4-digit birth year; for example, 1976)

* ③ How do you identify yourself?
☐ Male
☐ Female
☐ Other (please specify)

* ④ What state do you reside in?

* ⑤ What is the highest level of school you have completed or the highest degree you have received?
☐ Less than high school degree
☐ High school degree or equivalent (e.g., GED)
☐ Some college but no degree
☐ Associate degree
☐ Bachelor degree
☐ Graduate degree

Next ▶

Figure 36: Worker ID and Metadata User Sign In Page

Upon signing in and entering their metadata, each user would then submit their prediction for each of their games that week. Each user would answer the same type of questions as those seen in the NBA static survey, but instead of each user answering the same question as before, which is “how many points is your pick likely to win by?” the question was provided in the multiple choice question asking the user which team they believed would win, as shown in Figure 37.

Predict the outcome of the following games. (@ indicates the Home team. The starting pitcher for each team today is listed alongside the team name.)

NOTE: Today we are picking the games that will be played Tuesday, April 10th

* 6 Tampa Bay (Snell) @Chicago White Sox (Fulmer)

☐ Tampa Bay wins by 1

☐ Tampa Bay wins by 2+

☐ @Chicago White Sox wins by 1

☐ @Chicago White Sox wins by 2+

* 7 What percentage of the people taking this survey will pick the same winner you did?

0% 100%

* 8 You can bet \$0 to \$100 that the team you selected will win (regardless of number of runs). How much would you bet?

\$0 \$100

Figure 37: Example MLB Game Survey Each User Completed

After each user made a prediction for each match that week, they answered the same questions that were given in the NBA survey. Each user selected their *pick of the night*, or which team they thought was most likely to win that week, how much they were willing to bet that the team they chose would win, and what percentage of people taking this survey will get their pick of the night correct.

* 51 Pick of the Night: Of all your picks, which do you think is most likely to be correct?

☐ Tampa Bay @Chicago White Sox -- Tampa Bay wins	☐ NY Mets @Miami -- @Miami wins
☐ Tampa Bay @Chicago White Sox -- @Chicago White Sox wins	☐ LA Angels @Texas -- LA Angels wins
☐ Pittsburgh @Chicago Cubs -- Pittsburgh wins	☐ LA Angels @Texas -- @Texas wins
☐ Pittsburgh @Chicago Cubs -- @Chicago Cubs wins	☐ Houston @Minnesota -- Houston wins
☐ Detroit @Cleveland -- Detroit wins	☐ Houston @Minnesota -- @Minnesota wins
☐ Detroit @Cleveland -- @Cleveland wins	☐ Milwaukee @St. Louis -- Milwaukee wins
☐ NY Yankees @Boston -- NY Yankees wins	☐ Milwaukee @St. Louis -- @St. Louis wins
☐ NY Yankees @Boston -- @Boston wins	☐ Seattle @Kansas City -- Seattle wins
☐ Toronto @Baltimore -- Toronto wins	☐ Seattle @Kansas City -- @Kansas City wins
☐ Toronto @Baltimore -- @Baltimore wins	☐ San Diego @Colorado -- San Diego wins
☐ Atlanta @Washington -- Atlanta wins	☐ San Diego @Colorado -- @Colorado wins
☐ Atlanta @Washington -- @Washington wins	☐ Oakland @LA Dodgers -- Oakland wins
☐ Cincinnati @Philadelphia -- Cincinnati wins	☐ Oakland @LA Dodgers -- @LA Dodgers wins
☐ Cincinnati @Philadelphia -- @Philadelphia wins	☐ Arizona @San Francisco -- Arizona wins
☐ NY Mets @Miami -- NY Mets wins	☐ Arizona @San Francisco -- @San Francisco wins

* 52 You can bet \$0 to \$100 that the team you selected will win (regardless of number of runs). How much would you bet?

\$0

\$100

* 53 What percent of the people who take this survey will get their Pick of the Night correct?

0%

100%

◀ Prev

Next ▶

Figure 38: Example MLB Game Survey Each User Completed

After each user submitted all of their predictions for the week, they were asked to complete the same questions that were asked in the NBA survey as seen in Figure 39.

Thanks for making your picks! We're almost ready to take you to the Swarm where you and other MLB fans will pick those games collectively (as a "swarm")

Final questions:

54 What is your level of knowledge of the teams playing tomorrow?

1 - No knowledge at all 2 3 4 5 - Expert

55 What is your level of confidence in the picks you just made?

1 - Not confident at all 2 3 4 5 - Extremely confident

56 What percent of the games you picked today will you get right?

0% 100%

57 What percent of the games will the average person get right?

0% 100%

Figure 39: Example MLB Game Survey Each User Completed

Once the second prototype was working properly, the next dynamic survey iteration was created. The goal of this survey was to encompass all the information that was being captured in the static survey.

4.7 Final Dynamic Survey Iteration

The final iteration of the Dynamic Survey was similar to the second iteration except it encompassed all of the additional data collected in the static survey. As shown in Figure 40 and 41, each user would sign in with their Mechanical Turk Worker ID, as well as enter in metadata about themselves before entering the survey.

Sign In To Make Predictions

Please Enter the Following Information

What is your Mechanical Turk Worker ID?

In what year were you born (enter 4-digit birth year; for example, 1976)

How do you identify yourself?
☐ Male
☐ Female
☐ Other

What state do you reside in?

What is the highest level of school you have completed or the highest degree you have received?
☐ Less than high school degree
☐ High school degree or equivalent (e.g., GED)
☐ Some college but no degree
☐ Associate degree
☐ Bachelor degree
☐ Graduate degree

What username do you plan to use today?

Login

Figure 40: Example MLB Dynamic Survey Sign In Page Before Completion

Sign In To Make Predictions

Please Enter the Following Information

What is your Mechanical Turk Worker ID?

In what year were you born (enter 4-digit birth year; for example, 1976)

How do you identify yourself?
☒ Male
☐ Female
☐ Other

What state do you reside in?

What is the highest level of school you have completed or the highest degree you have received?
☐ Less than high school degree
☐ High school degree or equivalent (e.g., GED)
☒ Some college but no degree
☐ Associate degree
☐ Bachelor degree
☐ Graduate degree

What username do you plan to use today?

Login

Figure 41: Example MLB Dynamic Survey Sign In Page After Completion

After signing in and submitting metadata about themselves, each user would be shown instructions, would fill out their predictions for each game, and would be shown a summary of all their predictions. These pages were not changed from the previous iteration as seen in Figures 42-45

Instructions

If you make correct predictions you can win a \$50 bonus for these predictions!
Please read the following instructions before moving forward.

- 1) Your job is to predict the outcome of 15 upcoming MLB games.
- 2) You are expected to do research on each game before you make each prediction.
- 3) There are two ways to win, by predicting the correct Point Spread or by Game Wagering. First place in each competition wins a bonus of \$25, second place wins \$10, and third wins \$5. You can win both competitions for a total of \$50!

COMPETITION 1: POINT SPREAD

The three users with the smallest differences between their Point Spread predictions and the outcomes of this week's games win!

Example: If you predict the San Francisco Giants will win by 3 and they win by 2, your difference would be 1.

COMPETITION 2: GAME WAGERING

The three users that win the most virtual money by wagering on the matches this week win!

Example (see image below): If you wager \$96.00 for Kansas City to win and \$36.00 for Cleveland to win, you will win \$96.00 if Kansas City wins or \$36.00 if Cleveland wins.

Kansas City

\$96.00

Adjust slider to set your desired bets on each team

Cleveland

\$36.00

Get Started!

Figure 42: Example MLB Dynamic Survey Instructions Page

What Will Be the Outcome of the Mariners vs the White Sox?

Select by how much you think a team will win.

Mariners

+3 2

White Sox

+3 2

Select the probability that each team will win.

36%

64%

Using the distribution below, how much virtual money are you willing to wager that each team will win?
Note: You will win the amount below the team that wins.

\$62.79

\$84.79

Submit

Progress (0 of 15 predictions completed)

Figure 43: Example MLB Dynamic Survey Initial User's Prediction For a Game

Now Given the Crowd's Prediction (in Blue), What Do You Predict Will Be the Outcome? The Crowd's Prediction is the Average Forecast Across 100 Participants.

Select by how much you think a team will win.

Mariners | +3 | +1 | +1 | +3 | White Sox

Select the probability that each team will win.

57% | 43%

Using the distribution below, how much virtual money are you willing to wager that each team will win?
Note: You will win the amount below the team that wins.

\$79.75 | \$69.75

Submit

Progress (0 of 15 predictions completed)

Figure 44: Example MLB Dynamic Survey Final User’s Prediction For a Game

Thank You For Your Predictions!

1) Here is a summary of your predictions!

Matchup	Your Predicted Winner	Your Predicted Spread	Your Predicted Probability	Virtual Money Distribution
Mariners vs White Sox	White Sox	+1	71%	Mariners: \$51.00 White Sox: \$91.00
Cubs vs Indians	Indians	+1	67%	Cubs: \$59.04 Indians: \$87.04
Twins vs Yankees	Yankees	+1	69%	Twins: \$52.39 Yankees: \$90.39
Braves vs Reds	Braves	+1	64%	Braves: \$84.79 Reds: \$62.79
Diamondbacks vs Phillies	Phillies	+1	63%	Diamondbacks: \$61.56 Phillies: \$85.56
Tigers vs Pirates	Tigers	+1	66%	Tigers: \$89.76 Pirates: \$53.76
Rays vs Orioles	Orioles	+1	61%	Rays: \$49.59 Orioles: \$91.59
Red Sox vs Blue Jays	Red Sox	+1	62%	Red Sox: \$84.79 Blue Jays: \$62.79
Athletics vs Rangers	Athletics	+1	58%	Athletics: \$80.64 Rangers: \$68.64
Angels vs Astros	Angels	+2	65%	Angels: \$84.79 Astros: \$62.79
Brewers vs Royals	Royals	+1	66%	Brewers: \$56.44 Royals: \$88.44
Mets vs Cardinals	Mets	+3	72%	Mets: \$85.56 Cardinals: \$61.56
Padres vs Rockies	Rockies	+1	67%	Padres: \$61.56 Rockies: \$85.56
Marlins vs Dodgers	Dodgers	+2	72%	Marlins: \$57.75 Dodgers: \$87.75
Nationalists vs Giants	Giants	+1	75%	Nationalists: \$60.31 Giants: \$86.31

2) Out of all the predictions you have just made, which team do you think is the most likely to win and why? Try to be specific in your reasoning.

Figure 45: Example MLB Dynamic Survey Summary of User’s Predictions

As shown in the second portion of Figure 46, one question that I added to this Dynamic Survey iteration was, “What percentage of other participants will pick the same winner as you?” This

question was added to get a measure of how confident the user is that their favored team was the true favorite for the week.

2) Out of all the predictions you have just made, which team do you think is the most likely to win and why? Try to be specific in your reasoning.

Enter text here...

3) What percentage of other participants will pick the same winner as you?

0%

Submit

Figure 46: Example MLB Dynamic Survey Open Question Response and Percentage of Other Participants Will Pick the Same Winner Question

This final iteration of the Dynamic Survey also encompassed all the final questions that were in the Static Survey as shown in Figure 47.

Answer the Following Questions

What is your level of knowledge of the teams playing?

- ☐ No Knowledge
- ☐ Somewhat Educated
- ☐ Moderately Knowledgeable
- ☐ Very Knowledgeable
- ☐ Expert

What is your level of confidence in the picks you just made?

- ☐ Not Confident
- ☐ Somewhat Confident
- ☐ Moderately Confident
- ☐ Pretty Confident
- ☐ Very Confident

What percentage of the games you picked will you get right?

62%

What percentage of the games will the average person get right?

35%

Submit

Figure 47: Example MLB Dynamic Survey Final Confidence Questions

The feedback and final pages of the new survey were not changed from the last survey to the new survey as shown in Figure 48 and Figure 49.

Optional: Answer the Questions Below

Is there anything you think we should change about the entire questionnaire that you just completed? What was good? What was bad? Were there any parts that were confusing? Any input would be helpful!

Submit

Figure 48: Example MLB Dynamic Survey User Feedback Page

Figure 49: Example MLB Dynamic Survey Thank You Page

4.8 Machine Learning on Dynamic Survey Data

The goal of this project was to see how much better we could use this new survey to predict how well individual users would predict games given the data we collected during the Dynamic Survey (when compared to the data we collected during the Static Survey). Therefore, success was measured as the difference between my current prediction model's performance with the new data to that from data taken from the static survey. Furthermore, because the main focus was forecasting individual user predicting capability, all of the data that was collected was totaled and averaged for all of the predictions that a user would make for a given week.

For data analysis, the data was downloaded from the Google Firebase Database and converted from JSON files to .xlsx format. From the .xlsx files, each user's predictions were extracted via pandas' a Python data analysis library. Once in this format a variety of data was extracted. Initially data for every match each user made a prediction on was gathered and organized into a "MatchPrediction" object. The following information for each user's match predictions were used:

<u>Variable</u>	<u>Explanation</u>
team1	First team that was listed in the game
team2	Second team that was listed in the game
winner	The team that won the game
team1Score	How many runs the first team scored
team2Score	How many runs the second team scored
VegasProbPred	Probability that Vegas placed on the first team to win
initEstimateTime	Amount of time it took the user to submit their initial predictions once they were shown the prediction page
initSpreadPred	Initial user prediction of how much one team will win by
initT1ProbPred	Initial prediction of the probability that the first team will win
initT2ProbPred	Initial prediction of the probability that the second team will win
initT1MoneyPred	Initial amount of money the user is willing to place on the first team using the money slider
initT2MoneyPred	Initial amount of money the user is willing to place on the second team using the money slider
secEstimateTime	Amount of time it took the user to submit their final predictions after shown the "Crowd's Prediction"
secSpreadPred	Final user prediction of how much one team will win by
secT1ProbPred	Final prediction of the probability that the first team will win

secT2ProbPred	Final prediction of the probability that the second team will win
secT1MoneyPred	Final amount of money the user is willing to place on the first team using the money slider
secT2MoneyPred	Final amount of money the user is willing to place on the second team using the money slider
initPercentThatChoseTeam1 (For Support Score)	Percent of user's that week that initially predicted team 1 to win the match.
initPercentThatChoseTeam2 (For Support Score)	Percent of user's that week that initially predicted team 2 to win the match
secPercentThatChoseTeam1 (For Support Score)	Percent of user's that week that predicted team 1 to win the match on their final prediction
initPercentThatChoseTeam2 (For Support Score)	Percent of user's that week that predicted team 2 to win the match on their final prediction.

Table 1: Types of Data Gathered for Each Match

Because the purpose of this project was to predict each user's performance for the whole week (as opposed to individual predictions), all of the user's match predictions were summarized for the entire week and stored as totals and averages. Every data set consisted of a culmination of different users predicting a variety of matches for that given week. Therefore, the predictability of each week varied depending on how predictable the matches were for that week. Since the predictability varied from week to week, the average user's performance varied from week to week. In order to account for the change in match predictability across different weeks, the z score was calculated for each user depending on the performance of the other users that week.

The z score is calculated using the average, the standard deviation, and the samples value for each of the metrics being calculated. Once the average and standard deviation were found the z score was calculated by using the following equation [13]:

$$z\ score = \frac{x - \mu}{s} \quad (3)$$

where:

x = user measurement

μ = average for all users of that specific measurement for that week

s = standard deviation for all users of that specific measurement for that week

For example, in order to compare a single user's total money earned on their initial predictions, their total money earned on their initial predictions was compared to the average total money earned on initial predictions and total money earned on initial predictions standard deviation of each of the other players that week. By calculating the z-score on each significant piece of data, I was able to compare individual user's data and performance to all other user's data and performance regardless of what week each user was predicting matches.

Table 2 shows what data was collected and calculated for each user that completed all predictions for each survey.

<u>Variable</u>	<u>Description</u>
username	name the user used to identify themselves
date	day, month, and year the survey was taken
numOfGames	total number of games the user predicted in that survey
matchPredictions	list of all the matchPredictions the user predicted
initTotTimeTaken	total time taken to submit all of their initial

	predictions
initTotNumCorrect	total number of games the user predicted correctly on their initial prediction
initTotSpreadError	total number of runs the user was off on all of their initial spread predictions
initTotMoneyEarned	total amount of money the user earned from the money slider for all their initial predictions
initAvgTimeTaken	average time taken to submit each initial prediction per game
initAvgNumCorrect	average number of games the user predicted correctly on their initial prediction per game
initAvgSpreadError	average number of runs the user was off on for their initial spread predictions per game
initAvgMoneyEarned	average amount of money the user earned from the money slider for all their initial predictions per game
secTotTimeTaken	total time taken to submit all of their final predictions
secTotNumCorrect	total number of games the user predicted correctly on their final prediction
secTotSpreadError	total number of runs the user was off on all of their final spread predictions
secTotMoneyEarned	total amount of money the user earned from the money slider for all their final predictions
secAvgTimeTaken	average time taken to submit each final prediction per game
secAvgNumCorrect	average number of games the user predicted correctly on their final prediction per game
secAvgSpreadError	average number of runs the user was off on for their final spread predictions per game
secAvgMoneyEarned	average amount of money the user earned from the

	money slider for all their final predictions per game
initSupport	summation of the percent of users that agree with their initial predicted winner for each game
initOutlierScore	percent of games where user disagreed with 65% or more users on the winner of a game on first pred.
secSupport	summation of the percent of users that agree with their final predicted winner for each game
secOutlierScore	percent of games where user disagreed with 65% or more users on the winner of a game on final pred.
totDiffInFirstAndSecProb	total amount the user changed their probabilities from first to final predictions across all matches
avgDiffInFirstAndSecProb	avg. amount the user changed their probabilities from first to final predictions across all matches
totDiffInFirstProbAndVegas	total amount the user's first probability predictions differed from Vegas across all games
avgDiffInFirstProbAndVegas	average amount the user's first probability prediction differed from Vegas per game
totDiffInSecProbAndVegas	total amount the user's final probability predictions differed from Vegas across all games
avgDiffInSecProbAndVegas	average amount the user's final probability prediction differed from Vegas per game
totDiffInInitImpAndExpProb	total amount the user's first probability predictions and money slider predictions differed for every game
totDiffInSecImpAndExpProb	total amount the user's final probability predictions and money slider predictions differed for every game

Table 2: Types of Data Gathered for Each User

The z score for each of the variable shown in Table 2 was also calculated so that each user's data was comparable to every other user's data regardless of the week. Once data was collected for each user, I began data analysis. Each user's performance was measured based on the z-score of the average amount of money they earned per game. The average amount of money earned per game was used because it encompassed what team each user thought would win, as well as their confidence in their prediction. Unanimous AI engineers believe these characteristics are the best predictors of how well an individual performs in a swarm. Furthermore, the z-score was used so that user's performance from different weeks could be compared to each other. The amount of money earned by each user had a positive correlation between how many games the user predicted correctly and the z-score as shown in Figure 50. This makes sense considering users are more likely to place more money on teams they predict will win.

Correlation Coefficient: 0.492652047419

P-Value: 1.3101398183e-17

Correlation Coefficient: 0.464606067681

P-Value: 1.35377981598e-15

Figure 50: Graphs Comparing Z-Scores of Average Money Earned to Average Number of Predictions Correct

Conversely, the z-score of the average money earned per game showed was negatively correlated to the number of points the user predicted the spread incorrectly as shown in Figure

51. This is also intuitive because the more confident a user is in their prediction the less likely they are to be off on their spread prediction.

Correlation Coefficient:-0.549543242195

P-Value: 2.60020955963e-22

Correlation Coefficient: -0.50850865532

P-Value: 7.83512279913e-19

Figure 51: Graphs Comparing Z-Scores of Average Money Earned to Average Amount of Error in Spread Prediction

Before trying to predict individual user performance, the performances per game across all users were averaged. The average performance of all of the users across the first five weeks of data collection is shown in Table 3. The average performance of all of Vegas' predictions was also included in Table 3 for reference.

	<u>Initial Prediction Performance</u>	<u>Final Prediction Performance</u>	<u>Vegas Performance</u>
Average Spread Error	3.534	3.536	3.281
Average Correct Per Game	50.876%	50.763%	55.408%
Average Money Earned	\$70.33	\$70.26	\$74.48
Average Time Taken	13130.872ms	4354.0241ms	N/A

Table 3: Average User Performance for Initial Predictions and Final Predictions and Vegas Performance

Table 3 shows that a user was off by about 3.5 runs on average, predicting the winner of the game correctly approximately 51% of the time and earning approximately \$70.30 per game. Since a user can make \$75 per match without favoring one team over the other, using the money slider, and has a 50% chance of choosing the correct team without any prior information, this data shows that our users, on average, are not doing better than randomly guessing. One important detail to note, comparing the user's average initial prediction performance to their average final prediction performance yields interesting information. This data shows that the average spread error per game per user increased by +0.002 runs, correct winner prediction per game per user decreased by 0.113% per game, and average money earned decreased by \$0.07 dollars. This data shows that, on average, users were slightly better predictors on their initial predictions before the Crowd's Prediction perturbation. Moreover, by comparing the average time taken for initial and final predictions the data shows that on average each user takes 8776.848ms (8.777 seconds) longer on the first prediction than the second prediction. Table 3 also shows that Vegas performed better than the average user in every category but also did not perform well. Although the Vegas predictions predicted the winner 55% of the time, they made less than the average of \$75 for money earned by obtaining an average of \$74.48.

4.9 Data Analysis

The first step in analyzing the data was measuring, creating, and calculating data that was correlated with the user's performance. In a previous internal study done by an Unanimous AI engineer, Gregg Wilcox, user inputs for the static survey were related to users performance. From this study, the strongest correlation between user inputs into the static survey and user performance was a measurement called the Outlier Score and another measurement called the Support Score.

The Outlier Score was calculated by summing up the total number of games each individual user predicted against 65% of the other individuals in the swarm and average that total across the total number of games that the individual predicted. This measurement became a percentage representing how often an individual would predict against 65% percent of the population. The thought behind this measurement was that if a user predicted against the large majority a lot that individual was most likely guessing on the outcomes of the match without much knowledge. Moreover, individual who agreed with the majority frequently could have been well informed on their predictions and might have more knowledge than the average individual on some games that might cause them to predict differently than the large majority. Once the outlier score was calculated per user, it was converted into a z-score and compared to the performance of each user. Figure 52 shows how the user's performance relates to their outlier score.

Performance Based on Initial Predictions

Performance Based on Final Predictions

Figure 52: Average User Performance (Z Scores) Compared to Outlier Score (Z Score)

As shown in Figure 52, there is not a strong correlation between a user's outlier score and their performance.

The next measurement that was compared to each user's performance, was the support score. The support score was calculated by adding the percent of other users that agreed with the team choice that individual user picked to win. This calculation was then averaged across games to get a support score per game for each user. The support score when compared to individual user's performance can be seen in Figure 53.

Performance Based on Initial Predictions

Performance Based on Final Predictions

Figure 53: Average User Performance Compared to Support Score Per Game

Although the support score was more correlated than the outlier score, as seen in Figure 53 there is not a strong correlation between the user's performance and their average support score. The support score was the only data used in the internal study on the EPL to predict a user's performance. In the internal study, the support score was stored as a list of all of the users' support scores for every game. From this list of support scores, the kernel densities were calculated for each user and all placed on the same graph. Using the same code as the internal study (developed by Gregg Wilcox), I was able to plot the kernel densities for each user as seen in Figure 54. Every line in Figure 54 represents one user and the shape of the line represents an approximation of how often each user received a certain support score. The color of each line represents how well the user performed (z score of average initial money earned), where the top performers were more red and poor performers were more blue.

Figure 54: Kernel Densities for Support Scores

As seen in Figure 54, there does not appear to be a trend between how often a user received a certain support score and how the user performed.

One more piece of data that I investigated was how users performed compared to how much time they took to make predictions. I believed this might indicate a user's knowledge on the subject because they might be taking more time to do more research on the topic or thinking harder about their prediction. The correlation between user performance and how much time they took on average for each prediction can be seen in Figure 55.

Performance Based on Time Taken for
Initial Predictions

Performance Based on Time Taken for
Final Predictions

Figure 55: Average User Performance Compared to Time Taken Per Prediction

As shown in Figure 55, there is only a slight correlation between the amount of time taken to make the prediction and the user's performance.

The next piece of data that was investigated was the difference between the initial and final probability predictions. I hypothesized that if the user changed their prediction a lot after seeing the “Crowd’s Prediction” (actually the Vegas prediction), they probably did not know the Vegas probability before they made their initial prediction. Therefore, they probably do not know a lot about the topic and are not knowledgeable on their prediction. Figure 56 shows how the z score of absolute difference in probability between initial and final predictions relate to a user’s initial prediction performance and final prediction performance.

Performance Based on Initial Predictions

Performance Based on Final Predictions

Figure 56: Average User Performance Compared to Absolute Difference in Initial Probability Prediction and Final Probability Prediction

As seen in Figure 56, my hypothesis was incorrect, there was not much correlation between absolute difference in initial probability and final probability predictions.

The next data correlation I investigated was how the absolute difference in the user's probability prediction and Vegas's probability prediction. My hypothesis was that if a user was informed enough to know Vegas's probability on the match or knew enough to mimic the expert predictors in Vegas, they were a knowledgeable individual on the topic. Figure 57 shows how the z score of the absolute difference of a user's probability prediction and Vegas probability prediction affect a user's performance.

Performance of Initial Predictions

Performance of Final Predictions

Figure 57: Average User Performance Compared to Absolute Difference in User's Probability Prediction and Vegas's Probability Prediction

As seen in the bottom two graphs of Figure 57, the strongest correlation measured thus far is relating absolute difference in user's probability prediction and Vegas's probability prediction to the individual user's money earned. Since this had such a strong correlation, the kernel density was mapped for every user as seen in Figure 58 and Figure 59.

Figure 58: Kernel Density of Absolute Difference Between Each User's Initial Predicted Probability and Vegas Probability

Figure 59: Kernel Density of Absolute Difference Between Each User’s Final Predicted Probability and Vegas Probability

As seen in Figure 58 and 59 there is a trend between the top performers and how close their probability predictions are to Vegas. Top performers tend to have a small difference between their probability predictions and Vegas’s probability prediction.

The next piece of data that I investigated was the correlation between the average difference in each user’s implicit and explicit probabilities and their performance. When each user inputs their prediction into the probability slider they are explicitly predicting their belief of the probability either team winning. What each user does not know is that the money slider is actually working on the same scale as the probability slider, so when they input how much they are willing to wager on each team using the money slider, they are actually implicitly predicting another probability. If a user accurately reflects their probability prediction of one team winning they should match that same probability in their money slider. How this statistic correlates to the average individual is shown in Figure 60.

Performance of Initial Predictions

Performance of Final Predictions

Figure 60: Average User Performance Compared to Absolute Difference in User's Implicit Probability and Explicit Probability

As seen in Figure 60, the average difference in implicit and explicit probabilities has a relatively strong correlation to the average amount of money the user earned per game. Since this also had a relatively strong correlation, the kernel density was mapped for every user as seen in Figure 61 and Figure 62.

Figure 61: Kernel Density of Absolute Difference Between Each User's Initial Implicit Predicted Probability and Explicit Predicted Probability

Figure 62: Kernel Density of Absolute Difference Between Each User’s Final Implicit Predicted Probability and Explicit Predicted Probability

As seen in Figure 61 and 62 there is a trend between the top performers and how close their implicit and explicit probability predictions are to each other. Top performers tend to have a small difference between their implicit and explicit probability predictions.

Principal Component Analysis was also investigated but could not be implemented in this study because participants were predicting different matches every week. Therefore, the different matches each user were predicting could not be translated across different weeks.

Once the key pieces of data were calculated and extracted through data analysis, I used machine learning to predict how each user would perform from the answers they provided in the Dynamic Survey.

4.10 Predicting User's Performance With Machine Learning

The first way of measuring the performance of my machine learning method was by calculating the mean absolute error in the difference between the initial average money earned per game (z score) I predicted the user would make in comparison to the initial average money earned per game (z score) they actually made. The first step was creating a control measurement to compare my results with. In order to do this, I measured the mean absolute error of the difference between the actual average initial money each user earned to the average of all the users across all of the data I collected. This process was essentially predicting how every user would perform on average and then measuring the error between their performance and that average measurement. From this control method, I received a mean absolute error of 0.747.

In the internal study done on the EPL, the only input used to predict an individual user's performance was each user's support score. In the same study, each user's support score was kept as an array instead of being averaged per game. From this array format, each user's "kernel density" was calculated using the same code from the EPL internal study (developed by Unanimous AI engineer Gregg Wilcox). The kernel density was how often a user got a certain amount of support.

Using the kernel densities for the support scores from eight different weeks of data collection and 301 user predictions, I was able to mimic the form of the input data that was used for the internal EPL study (data shown in Figure 54). Using the kernel densities for support scores as input data I was able to train and test different machine learning techniques.

Each machine learning model was trained on 80% of the data and tested on 20% of the data. This process of training and testing was repeated and the absolute mean error was averaged using different data to train and different data to test to ensure that no outliers in the data drastically changed the results. Three different machine learning models were used and implemented with a python library called "scikit-learn": Linear Regression, K Nearest Neighbors, and Random Forests [15]. Because a primary concern was to get results quickly so that Unanimous AI could make decisions about their surveys quickly, scikit-learn and these machine

learning models were used because of their ease of use. More specifically, these machine learning models were advised for quick regression modeling by the engineers at Unanimous AI. Furthermore, the Random Forest and Linear Regression models used the default scikit-learn setting for regression while the K Nearest Neighbors used all the default settings other than the number of neighbors (which was set to 50 after trial and error). The average absolute mean error using each of these models is shown in Table 4.

<u>Machine Learning Model</u>	<u>Average Absolute Mean Error</u>
Random Forest	0.788
Linear Regression	0.778
K Nearest Neighbors	0.769
Control (for comparison)	0.747

Table 4: Machine Learning Model and Performance for the Kernel Densities for Support Scores as Input

As seen in Table 4, using the kernel densities for support score as the only input to these machine learning models predicts less effectively than our control group. Using this technique actually results in less accuracy than taking the average output of the training data.

Next, I used all of the data that I collected and calculated to predict each user's average initial money earned. I used the z scores of the following data:

1. Average total difference in first and second probability predictions
2. Average time taken to make first predictions
3. Average time taken to make final predictions
4. Initial support score
5. Final support score
6. Initial outlier score

7. Final outlier score
8. Average difference in initial probability prediction and Vegas probability prediction
9. Average difference in final probability prediction and Vegas probability prediction
10. Average difference in initial implicit and explicit predictions (difference in money and probability sliders)
11. Average difference in final implicit and explicit predictions (difference in money and probability sliders)

<u>Machine Learning Model</u>	<u>Average Absolute Mean Error</u>
Random Forest	0.713
Linear Regression	0.706
K Nearest Neighbors	0.695
Control (for comparison)	0.747

Table 5: Machine Learning Models and Performance Using All Data Collected and Calculated From This Study

As seen in Table 5, using this new data to train the same models, the machine learning models were able to outperform the control prediction by up to 6.6%. Furthermore, I was able to improve the models by removing both the initial and final outlier scores and both the initial and final support scores leaving the following data to train with:

1. Average total difference in first and second probability predictions
2. Average time taken to make first predictions
3. Average time taken to make final predictions
4. Average difference in initial probability prediction and Vegas probability prediction
5. Average difference in final probability prediction and Vegas probability prediction
6. Average difference in initial implicit and explicit predictions (difference in money and probability sliders)

7. Average difference in final implicit and explicit predictions (difference in money and probability sliders)

Training and testing with this data, I got the results shown in Table 6.

<u>Machine Learning Model</u>	<u>Average Absolute Mean Error</u>
Random Forest	0.715
Linear Regression	0.703
K Nearest Neighbors	0.685
Control (for comparison)	0.747

Table 6: Machine Learning Models and Performance Using All Data Besides Outlier and Support Scores

As seen in Table 6, the average absolute mean error was reduced by 8.3% for the K Nearest Neighbors.

Limiting the input data to the top four highest correlations (minimum correlation of .35), I used the following as inputs:

1. Average difference in initial probability prediction and Vegas probability prediction
2. Average difference in final probability prediction and Vegas probability prediction
3. Average difference in initial implicit and explicit predictions (difference in money and probability sliders)
4. Average difference in final implicit and explicit predictions (difference in money and probability sliders)

<u>Machine Learning Model</u>	<u>Average Absolute Mean Error</u>
-------------------------------	------------------------------------

Random Forest	0.693
Linear Regression	0.691
K Nearest Neighbors	0.674
Control (for comparison)	0.747

Table 7: Machine Learning Models and Performance Using Most Correlated Data

As seen in Table 7, by reducing the amount of training data I was able to further increase the accuracy of the machine learning models. When comparing these mean absolute errors to the control mean absolute error I was able to decrease the mean absolute error by 10%.

The whole point of predicting how well users will perform is so that the top performing users are let into the swarm. Using these machine learning methods, I ordered users for a given week based on how well each method predicted they would perform. By only taking the predicted top performers for any given week, I was able to change the average performance (average initial money earned) of the group. The blue lines in Figure 63 show the average performance of the group when compared to how many of the top performers are taken from the group for the first five weeks of MLB data collection and the orange line shows the average performance of all users for that week. As shown in Figure 63-68, I first used the different machine learning methods to order the participants using the data that was previously used to predict user performance (outlier score and support score).

Figure 63: Performance of Top Predicted Performers Using Outlier and Support Score (Linear Regression Model) By Number of Users Per Week

Using the five weeks' worth of data seen in Figure 63, I measured the average in change in user performance across the five weeks shown in Figure 64.

Figure 64: Average Performance of Top Predicted Performers Using Outlier and Support Score (Linear Regression Model) By Number of Users

As seen in Figure 63 and 64, there was not a substantial increase in average user performance using the Outlier and Support Score's and the Linear Regression model. This process was repeated using the K Nearest Neighbors and Random Forests models as shown in Figure 65-68.

Figure 65: Performance of Top Predicted Performers Using Outlier and Support Score (K Nearest Neighbors Model) By Number of Users Per Week

Figure 66: Avg. Performance of Top Predicted Performers Using Outlier and Support Score (K Nearest Neighbors Model) By Number of Users

Figure 67: Performance of Top Predicted Performers Using Outlier and Support Score (Random Forest Model) By Number of Users Per Week

Figure 68: Average Performance of Top Predicted Performers Using Outlier and Support Score (Random Forest Model) By Number of Users

As seen in Figures 65-68, using the outlier and support score with the three machine learning techniques did not amplify the average performance of the group.

Using the same three machine learning models and the same ordering methodology, but supplementing the data which gave the most accurate mean absolute error, I received much better results as seen in Figures 69-74.

Figure 69: Performance of Top Predicted Performers Using Most Correlated Data (Linear Regression Model) By Number of Users Per Week

Figure 70: Average Performance of Top Predicted Performers Using Most Correlated Data (Linear Regression Model) By Number of Users Across the First Five Weeks

As shown in Figure 69 and 70, by using the more correlated data and the Linear Regression Model, I was able to increase the average performance of the group by up to 5.5% percent. This process was then repeated using the same input data to train and test the K Nearest Neighbors Model and Random Forest Model, as seen in Figures 71-74.

Figure 71: Performance of Top Predicted Performers Using Most Correlated Data (K Nearest Neighbors Model) By Number of Users Per Week

Figure 72: Average Performance of Top Predicted Performers Using Most Correlated Data (K Nearest Neighbors Model) By Number of Users Across the First Five Weeks

Figure 73: Performance of Top Predicted Performers Using Most Correlated Data (Random Forest Model) By Number of Users Per Week

Figure 74: Average Performance of Top Predicted Performers Using Most Correlated Data (Random Forest Model) By Number of Users Across the First Five Weeks

As seen in Figures 69-74, the average performance is amplified most across these 5 weeks of MLB data using the highest correlated data and the Linear Regression model.

Chapter 5

CONCLUSION

This study provides compelling evidence to suggest that the proposed method and metrics perform significantly better than the pre-existing ones. The new dynamic survey captures more informative data on a user's knowledge regarding a topic when compared to the previous survey. Thus, by gathering data on each user's predictions as well as how they input those predictions, I was able to gather data that was 5 times more correlated to a user's performance than the previous input data used (support score). Also, utilizing the data extracted from my new survey, the proposed method reduced the mean absolute error by approximately 10% on predicted user performance when compared to the control group and approximately 12% percent when compared to the previous methodology. This method also further increased the average performance of the group by 5.5% by sorting the performers best to worst and limiting the number of users participating. Although the results are limited to MLB for this study and could differ from sport to sport, the data correlation and machine learning techniques proposed in this study prove that the new dynamic survey captures more informative data than the previous survey.

This study had various strengths and limitations unique to Unanimous AI. Advantageously, Unanimous AI was interested in implementing the results of this new survey and were willing to pay individuals to take the survey. Other researchers may not be able to gather as much feedback in the same amount of time without this financial motivation for participants. Therefore, I had access to a sufficient data relatively quick. Detrimentally, Unanimous AI was making decisions on a week-to-week basis, and is in their early stages as a company so I did not have time as an abundant resource. Therefore, extensive research for every design decision was not as important as it might have been otherwise because time resources were limited.

In conclusion, this study focused on creating a new survey that would produce metrics that were more indicative of a user's performance when predicting the MLB. The results provide strong evidence that the new dynamic survey is superior to the previous survey method.

However, the scope of the data analysis and machine learning methods that have been used is still limited; producing results quickly was a high priority so that the company could make production decisions. Therefore, there still remains a lot of room for investigation into higher correlated data as well as more suitable machine learning techniques.

Chapter 6

FUTURE WORK

There are several areas related to this study I recommend to anyone looking to pursue this project further. First, creating individual swarms for each game instead of the same swarm predicting every game may prove as a useful method to further increase wisdom within a swarm. Currently, one swarm answers questions on every game for a sport for a given week. Removing users based on their knowledge for specific games for a week and place those users into individual swarms that answer questions solely based on each game they could increase the performance of their swarms.

I also believe that there are several methods that could potentially optimize the prediction of a user performance. For example, the machine learning models could be trained with how predictable each of the weeks was as additional information such that weeks that were more predictable would hold a heavier weight in the training than those that were less predictable. Additionally, training a machine learning model on the open feedback the user's implemented, as well as on the metadata that was collected on each user could lead to more accurate predictions. Finally, this same study could also be done on different sports, with different point spreads and different certainties.

BIBLIOGRAPHY

- [1]M. Dorigo and M. Birattari, "Swarm intelligence", *Scholarpedia*, 2018. [Online]. Available: http://www.scholarpedia.org/article/Swarm_intelligence. [Accessed: 29- May- 2018].
- [2]L. Rosenburg, "Human Swarms, a real-time paradigm for Collective Intelligence", 2018.
- [3]R. Axelrod, *The Evolution of Cooperation*. 2018.
- [4]L. Muchnik, S. Aral and S. Taylor, *Social Influence Bias: A Randomized Experiment*. Science Magazine, 2013, pp. 647-651.
- [5]"WHAT IS SWARM AI? - UNANIMOUS AI", *UNANIMOUS AI*, 2018. [Online]. Available: <https://unanimous.ai/what-is-si/>. [Accessed: 29- May- 2018].
- [6]A. Kay, "Kentucky Derby Partnering With A.I. Company That Correctly Predicted Last Year's Superfecta", *Forbes.com*, 2018. [Online]. Available: <https://www.forbes.com/sites/alexkay/2017/05/03/kentucky-derby-partnering-with-a-i-company-that-correctly-predicted-last-years-superfecta/#13b5ac82abaa>. [Accessed: 29- May- 2018].
- [7]"Unanimous a Finalist for AI Innovation of the Year at SXSW - UNANIMOUS AI", *UNANIMOUS AI*, 2018. [Online]. Available: <https://unanimous.ai/unanimous-finalist-ai-innovation-year-sxsw/>. [Accessed: 29- May- 2018].
- [8]"Unanimous AI Press & media coverage", *UNANIMOUS AI*, 2018. [Online]. Available: <https://unanimous.ai/press-coverage/>. [Accessed: 29- May- 2018].
- [9]G. Wilcox, "EPL Crowd Curation", Unanimous AI, 2018.
- [10]A. Ven and A. Delbecq, "The Effectiveness of Nominal, Delphi, and Interacting Group Decision Making Processes", *Academy of Management Journal*, vol. 17, no. 4, pp. 605-621, 1974.

- [11]A. Sarkissian, "Different Techniques in Group Decision-Making", *Yourbusiness.azcentral.com*, 2018. [Online]. Available: <https://yourbusiness.azcentral.com/different-techniques-group-decisionmaking-17366.html>. [Accessed: 29- May- 2018].
- [12]"Brier Score: Definition, Examples", *Statistics How To*, 2018. [Online]. Available: <http://www.statisticshowto.com/brier-score/>. [Accessed: 29- May- 2018].
- [13]"Z-Score: Definition, Formula and Calculation", *Statistics How To*, 2018. [Online]. Available: <http://www.statisticshowto.com/probability-and-statistics/z-score/>. [Accessed: 29- May- 2018].
- [14]"Mean Error: Definition", *Statistics How To*, 2018. [Online]. Available: <http://www.statisticshowto.com/mean-error/>. [Accessed: 29- May- 2018].
- [15]"scikit-learn: machine learning in Python — scikit-learn 0.19.1 documentation", *Scikit-learn.org*, 2018. [Online]. Available: <http://scikit-learn.org/stable/>. [Accessed: 29- May- 2018].