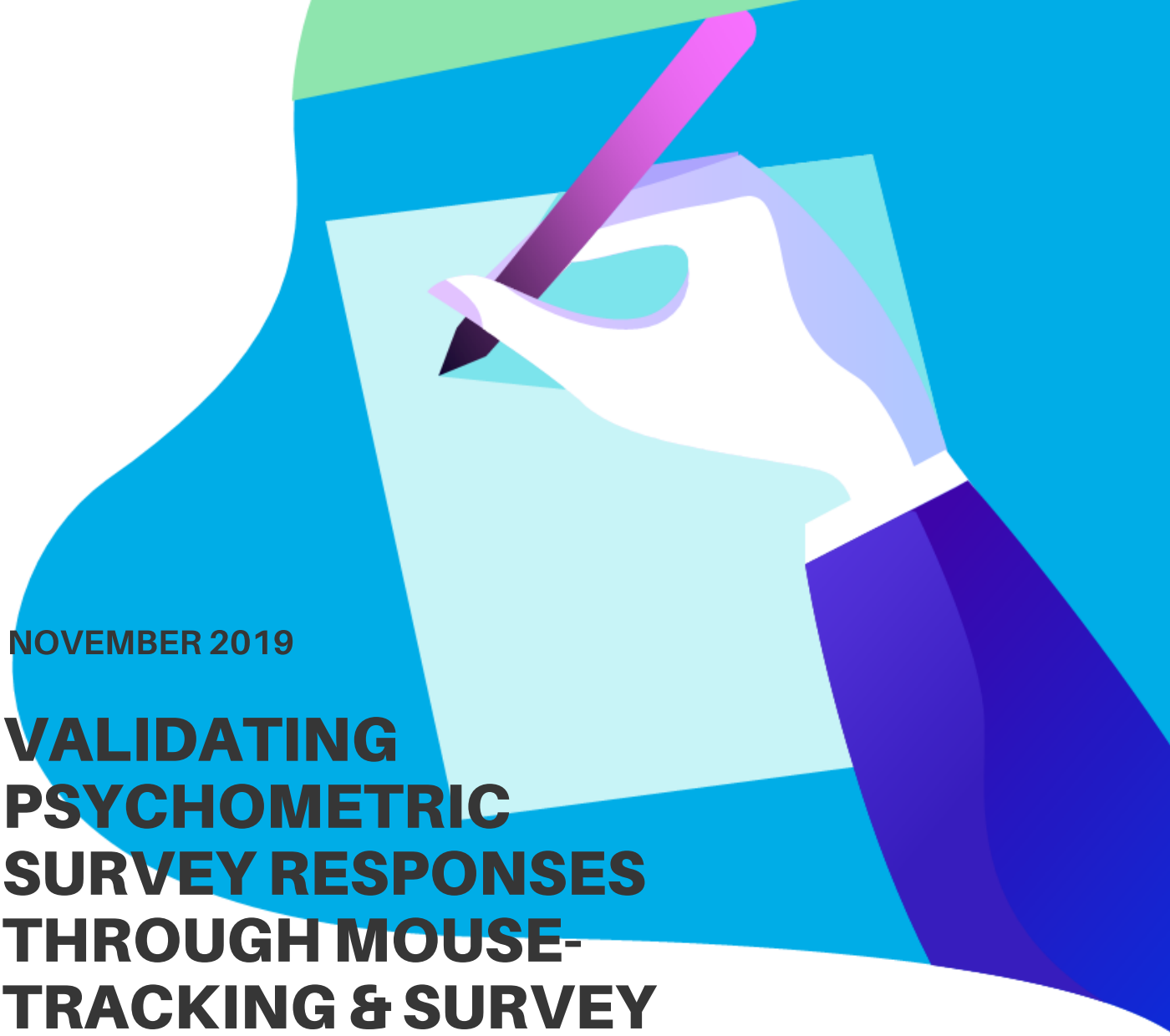


NOVEMBER 2019

VALIDATING PSYCHOMETRIC SURVEY RESPONSES THROUGH MOUSE-TRACKING & SURVEY ANALYTICS TECHNIQUES

Columbia University

x

Dotin.us

Team Alpha
Alberto Mastrotto, Anderson Nelson, Dev Sharma,
Ergeta Muca, Kristina Liapchin, Luis Losada, Mayur Bansal

Intro

Throughout this research project, our team has examined ways that dotin can distinguish between legitimate and false responses to their psychometric surveys. These psychometric survey responses constitute a crucial part of dotin's AI engine that captures the accurate digital personality fingerprint of users and helps assess their personality profiles. Therefore, the fact that dotin found 30% of responses to their psychometric surveys to be false is an alarming finding, which can drastically harm the accuracy of the AI engine at the core of the company's business, if disregarded. Based on this, our project was aimed at improving the existing validation method for user responses to dotin's psychometric surveys. To focus our efforts, we explored the following research questions:

Does the level of suspicious behavior vary across different survey questions?

How do we use user mouse activity to validate answers to survey questions?

Features

Based on these questions, our team worked on creating additional features from the original data, which were pivotal to our modeling efforts and fell into the following key categories:

Time

Screen coverage

Distance traveled

Direction of movements

Survey-focused

The basis of creating these features was on literature review of mouse path analytics as well as common business knowledge. Although not all features ended up being used in our final models, they played a big role in our exploratory data analysis and in developing our models to help us get the best and most accurate results. A detailed table containing all features created and used in modeling can be found in **Figure 1** in the Appendix.

Based on our initial exploratory data analysis, we proceeded with building a few different models to help us identify fraudulent survey responses with the goal of improving the current validation method used by dotin.

Validation Approach-es

We developed the following three methods throughout the course of this project:

1. ***Rule-based Approach***
2. ***Long Short-Term Memory***
3. ***Hidden Markov Model***

We decided to use these three approaches to compare how the different methods would perform, considering the lack of accurately labeled data in the original dataset.

That way, we would be able to make better and more well-informed recommendations for dotin with regards to a new validation method to use for their psychometric survey responses.

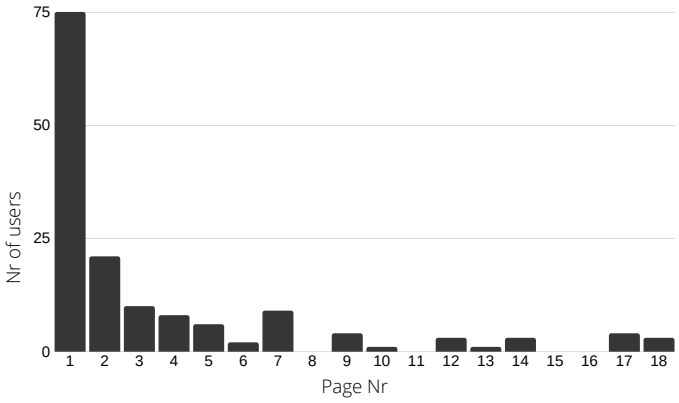
RULE-BASED APPROACH

From our EDA we identified that the tracking method used to generate the mouse path dataset presented some challenges as many of the user's paths weren't fully recorded. Out of the 755 user's data, only 54 fulfilled the basic requirement of clicking the 196 radio buttons pertaining to individual questions. Therefore, we added our data collection recommendations in the final section of our paper.

Before diving into the modeling, our team found essential to create alternative ways to flag anomalous users other than dotin's current validation method. In order to generate such features, we used both common business sense and advanced outlier detection techniques that allow us to understand each user from different angles. Such features will serve as a way to validate dotin's current validation method as well as allow us to generate basic business rules to flag suspicious behavior. Some of these features will then be used to test our models.

Anomalies by Score

From our analysis, we discovered that 150 of the 755 users surveyed answer at least one page of the survey with all of the same scores. We then assume that there is no page where such an event would be plausible, therefore these users are flagged as anomalous.



Count of users with same answer across pages

Anomalies by Time

We then proceeded to focus on the time perspective by estimating the read time that an honest user would take to read the survey and compared it with the actual completion time taken by each individual user. The benchmark read time of a regular user was derived from **Medium's** read-time algorithm, which is based on the average reading speed of an adult (~256 wpm). The read time was calculated for all the individual questions in our users' surveys and compared to the time it took them to click one radio button to another (an indication of them moving from one question to the other). From our analysis, on average, a user that completed the entire survey would need 5 minutes and 30 seconds to at least read all the 196 questions, yet 33% of our surveyed users took less time than that. Therefore, we flagged users that take less than the calculated reading times as anomalous.

Anomalies by Topic

Finally, we focused on the first 40 questions of the survey to create our own topics and scored each user based on how they deviate in answering the survey questions. For each topic, we aggregated questions that are either positive or negative (i.e. Tidy/Untidy) and we analyzed how users answer differently for similar questions. The underlying assumption is that if the user is deviating from their answers each time, this indicates that he/she is not fully paying attention to the questions. Questions with opposite behavioral traits should then present scores that are opposite (low standard deviation). In our analysis, we chose a threshold for a standard deviation of 2 to identify unfocused users, consequently resulting in 33% of users answering opposite questions with similar answers, (i.e. Tidy = 5; Untidy = 5). Based on this analysis, such users will then be flagged as suspicious.

Aggregated Flag Scores based on the Rule-based Approach

In order to identify our suspicious users based on these 3 features, we assign a flag score to each user. This flag score indicates the level of suspicion that our rule-based approach suggests. The value of the flag score ranges from 0 to 1 where 0 indicates that the user can be validated and a value greater than 0.33 means that the user appears as an anomaly in 2 or more features which definitely suggests that the user is a red flag. Based on our results, the model identifies 46 users i.e. (7 percent of the total users) in this list of flag users.

Generating new validation labels using Autoencoders

Due to the outlier detection nature of the features explained above, we decided to take an unsupervised learning approach to create a new validation method. We used an outlier detection algorithm to create our own labels of valid and non-valid users. The nature of our data set then required an approach that could deal with many variables but few observations (704 observations that represent features for each user).

Training the Autoencoder

In order to train our autoencoder we handpicked 144 users that based on our analysis had completed the entire survey and whose mouse activity data was clean. The model trained with these users had 25 Neurons on the input and output layers and two hidden layers of 2 neurons each. The compression used a sigmoid activation function and the mean squared error of the process was 11.49. The results showed that 76% of the users were classified as non-outliers while the rest were classified as outliers. Although this method did take into account mouse behavior, we wanted to focus on mouse movement at a more granular level. We further use the output of this validation method as a dependent variable in our LSTM model.

LONG-SHORT TERM MEMORY (LSTM) APPROACH

Recurrent Neural Networks (RNN) have grown to be a popular tool in Natural Language Processing for Language Modeling. Hence, RNN implementations are no strangers to sequence-based applications. As in language modeling, an RNN is responsible for predicting the next token. Our approach to applying RNNs to the problem at hand consists of two key stages:Training a model that can predict a user's next movement. Transferring the learning from the first model to a classifier model for predicting survey response validation trained using autoencoders.

Data Preparation

In order to feed an RNN, we needed to transform our data into a sequential format that the RNN can understand. For this purpose, we created string-based tokens which identified the cardinal directions and magnitudes of a user's movements. Page changes are identified with the "pagechange" token. All of a user's movements were appended to a single tokenized list of strings. For example, a user's movements might start off as ["nw", "1", "sw", "3" .... "pagechange" "ne", "2"]. For memory efficiency, movements were

averaged out between radio clicks. Since our RNN's loss function would be Cross-Entropy instead of Mean Squared Error, we scaled the magnitudes significantly to create large bins. This means that if our model predicts "8" as a magnitude whereas it should have predicted "7" for example, it is justified to penalize the model just as if it would have predicted a "2" because a one-point shift in magnitude is quite significant. Lastly, we split our data into training and validation sets based on a 70:30 split respectively.

Model Architecture

We used LSTM as a model as they are robust against the vanishing gradient problem. Similar to RNNs, our models carried two types of parameters: token embeddings and hidden states. Weights also included those which the LSTM uses to determine how significant of an adjustment should be made for the new sequential input. Tokenized user movements were inputted in mini-batches of 8 and trained on a 6GB 1070 GPU. For batches of a user where input length differed, padding was added to the end of shorter sequences. We used the cross-entropy loss function, and the evaluation metric for both the language model and the classifier was Accuracy.

Once the first model was trained, we replaced the final linear layer with a classification head of N x 2 dimensions, which produced a binary label where N is the input dimensions of the final hidden state from our LSTM.

Model Results

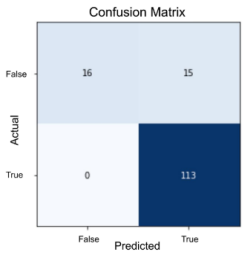
In stage 1 of the language model, we trained the model on our training set. We achieved the following results in predicting the next token on our validation set:

epoch	train_loss	valid_loss	accuracy	time
0	2.336163	1.453722	0.521071	00:09
1	1.721388	1.280523	0.577649	00:09
2	1.470130	1.229108	0.586503	00:09
3	1.350252	1.212001	0.588408	00:09
4	1.275603	1.179907	0.595714	00:09
5	1.218212	1.168359	0.584360	00:09
...	...	...	...	...
20	0.956453	0.997654	0.638661	00:09
21	0.945954	0.995324	0.638824	00:09
22	0.944055	0.996780	0.639673	00:09
23	0.939271	0.996102	0.640015	00:09
24	0.938368	0.995273	**0.639896**	00:09

We received an accuracy of ~64% after twenty-five epochs. Now that we have developed a model that was able to predict the next word, we removed the language model head and replaced it with a classifier head with randomly generated parameters. Hence, we trained this head to classify the validation status of surveys. Following are the results of predicting the survey validation status after five epochs:

epoch	train_loss	valid_loss	accuracy	time
0	0.716256	0.499177	0.881944	00:16
1	0.649158	0.408814	0.840278	00:16
2	0.583171	0.345984	0.902778	00:16
3	0.524735	0.313288	0.895833	00:16
4	0.498957	0.332085	**0.895833**	00:16

Our LSTM model produced a **~90%** accuracy on predicting whether a user's survey response is valid or invalid. Following is the confusion matrix and the classification report:



	precision	recall	f1-score	support
0	0.58	0.90	0.71	20
1	0.98	0.90	0.94	124
accuracy			0.90	144
macro avg	0.78	0.90	0.82	144
weighted avg	0.93	0.90	0.90	144

This approach produced the highest recall. This means that this model was the best at catching the most amount of invalid surveys identified by the autoencoder.

The LSTM approach was able to produce strong results, and it can certainly be used in an ensemble of multiple models to prevent general overfitting. Given its efficient runtime and high accuracy, we can also recommend it as a model of choice to predict autoencoder based labels if restrictions are posed. However, we ultimately stand by that the most generalizable results are achieved using a combination of approaches.

HIDDEN MARKOV MODEL

Our third proposed method to determine the users' authenticity in survey responses is by analyzing the sequence of user movement using a Hidden Markov Model (HMM). HMM is an approach to model sequential data, and implies that the Markov Model underlying the data is unknown. Probabilistic graphical models such as HMM have been successfully used to identify user web activity. For such models, the sequences of observation are crucial for training and inference processes. We made a series of assumptions and data transformations, and we will provide an overview of the steps to produce the model and summary results and findings.

Data Transformation

We converted the window aspect ratios into device types and discovered that certain users elected to take the survey on a laptop or mobile device. We believe that the movement patterns observed by people on mobile devices differ from those on a laptop. We solely focused on users who completed the survey using a laptop for modeling purposes.

We focused on users' coordinates across the survey duration and discovered that there's a lot of noise in the movements. To run an effective model, we converted the coordinates into discrete observations representing cardinal directions. For instance, a

movement to the right of the x-axis and up on the y-axis is labeled as North East. In total, nine labels were created: North East, North West, North, South East, South West, South, West, East, and No Movement. Using these directions as states S, we create a sequence of observations concerning mouse movement activity by observing a user as they complete the survey. The priority is to understand the overall direction of the user movement.

We recognize that users are navigating through survey pages, so we use the coordinates of the next button to estimate when each user moves to the next page. After analyzing each survey page, we realized that each user has a unique layout and the mouse path that users exhibit varies. Furthermore, considering that the number of mouse movement records varies per page, we decided to analyze the first 200 observations per user. We also removed the users that took the survey multiple times. After multiple attempts those users have become accustomed to the survey design and movement would be based on memory.

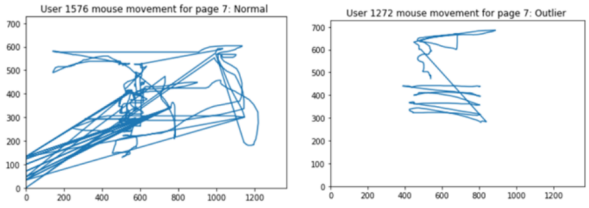
Only 66 users met the defined criteria for further analysis in this approach. We trained the HMM using the Baum-Welch algorithm to estimate the transition matrix, state distribution, and output distribution. We train the algorithm to recognize the patterns in each page and apply the forward algorithm to calculate the observation log probability of each observed user sequence per page. A low log probability is interpreted as having a less likely occurrence:

	page 1	page 2	page 3	page 4	page 5	page 6	page 7	page 8	page 9	page 10	page 11	page 12	page 13	page 14	page 15
384	0.548	0.533	-0.022	-0.714	0.648	0.818	0.795	0.326	0.618	0.675	0.617	0.784	0.900	0.776	0.700
422	-2.184	-5.829	-4.181	-4.445	-4.693	-4.275	-6.126	-4.628	-4.139	-1.509	-2.916	-1.492	-0.605	-0.314	-0.380
448	0.563	0.949	0.500	-0.185	1.068	1.426	0.280	0.532	0.619	0.539	-0.148	0.667	-0.477	0.491	0.129
507	0.567	0.458	-0.333	0.172	-0.050	0.022	0.274	-0.681	-0.240	0.947	0.867	0.978	1.140	1.175	0.667
549	0.202	0.890	0.746	0.769	0.563	1.001	0.930	0.746	0.888	0.676	0.720	-0.408	-1.199	-1.342	-2.574

We scale each observation and apply an isolation forest to identify suspicious users. Out of the 66 users, 11%, or 7 users were labeled as suspicious.

User Id	422	727	866	1272	1297	1314	1495

We compare two users for page 7 to illustrate their mouse movements. User 1576 movements move across the entire page while user 1272 movements are targeted and deliberate.



Assumptions and Limitations

The accuracy of the HMM is dependent on the validity of the assumptions, and the quality of the data. We therefore identify the assumptions and limitations of this approach:

1. The captured data doesn't distinguish when users are using their mouse to complete the survey vs browsing the internet.
2. The model assumes that the majority of users are completing the survey in good faith. If most users are falsely completing the survey, then the users that are attempting to complete the survey in good faith will be flagged.
3. The model was trained on the first 200 sequential observations, and user's patterns could differ as they progress through the pages. There are some users with 15,000 observations. Using an analogy, we are assuming that we can predict whether someone will win a 100m race using the first 10m.
4. The page labels were estimated using the coordinates of the next button on each page. Those labels represent our best estimate and may not truly reflect when the user page changes.

Findings

Despite the efficiency of such a probabilistic graphical model in segmenting and labeling stochastic sequences, its performance is adversely affected by the imperfect quality of data used for the construction of sequential observations. While the HMM can be useful in providing the probability of sequence, due to the quality of the data it shouldn't be the sole source. Therefore, we would suggest using a combination of methods in order to identify invalid survey responses.

Conclusion

To conclude, we have developed three different methods to validate psychometric survey responses for dotin. These three methods helped us answer our initial research questions, in particular:

Does the level of suspicious behavior vary across different types of survey questions?

From our outliers section, we were able to create general business rules to help us identify user behavior across pages:

1. Users that use the **same scores across a single page** can be flagged as suspicious.
2. Users that take **more than 5:30 minutes** to answer the survey can be flagged as suspicious.
3. Users that score **above a standard deviation of 2** in our topic modeling, will be flagged as suspicious.

It is important to highlight the importance of having such business rules in the identification of suspicious behavior as flagging users could be an easy to implement rule approach to validating surveys. We envision this method to become the first line of defense from suspicious users, and an easy to implement solution to flag suspicious behavior across each page, and ultimately, the entire survey.

How do we use user mouse activity to validate survey answers to psychometric questions?

Through this analysis, we are looking to gain a better understanding of the user journey throughout the survey. The goal is to see if different ways of interacting with the survey could be a baseline to create a model that through direction and magnitude of mouse movement would help us identify whether a user is correctly filling out the survey. To tackle the question we used both supervised and unsupervised techniques:

Supervised Learning: LSTM

We implemented an autoencoder to generate a new validation label, independent of dotin's current approach. We then used this variable as labels in an LSTM model that can classify suspicious user behavior.

Unsupervised Learning: HMM

We used a probabilistic approach that analyzed the sequence of user movement with the Hidden Markov Model and complemented it with the Isolation Forest Algorithm to find the number of suspicious users.

Putting together our findings, we can now compare the performance and results generated by the three different methods:

	Rule-based Approach	LSTM	HMM
Method	Rule-based	Supervised	Unsupervised
Training time	NA	~6 Minutes	~1 Hour
Percentage of Users Tested	94%	44%	9%
Percentage of Suspicious Users	44%	22%	11%

As we can see, each model was trained on a different set of users due to the limitations we faced with the quality of the original data. Therefore we would not recommend using one single model at this point, yet we could proceed with a hybrid approach that takes into consideration 3 models to validate users. We believe that an improved data collection method will further help improve the results of the individual models, as well as the overall hybrid model, enabling dotin improve the accuracy of their validation method for psychometric survey responses.

Recommendations

Data Collection Methods

In our EDA, we were able to identify only 54 users who fulfilled the basic requirement of clicking the 196 radio buttons pertaining to individual questions. This poses a huge threat to our modeling efforts as it shows how the current data collection method failed to capture the complete path a user takes when answering the survey. In order to understand the underlying problem, we tested a variety of methods and found that users are allowed to use their keyboards as they answer through the survey. In fact, by using the "Tab", "Space", and "Arrow Keys" commands, a user can theoretically answer the entire survey without the need to use their mouse. The fact that Mechanical Turks are used to answering many surveys throughout the day, also suggests that they got used to filling out surveys through their keyboard, as this tends to be very efficient.

Our recommendation consists of testing a new batch of users to fill out the survey without the option of using their keyboard. The test will prove to be successful once the number of radio buttons in the survey is equal to or greater than the 196 questions required to finish the survey. In the case that this doesn't prove to be true, then we recommend looking into a different library to track mouse movements, clicks, and scrolls.

New Validation Framework

Considering the current state of dotin's data collection, we would recommend proceeding with a hybrid validation method for psychometric survey responses that consists of the three different validation methods that we have developed throughout the course of this project:

1. Rule-based Approach
2. Supervised LSTM - based Approach
3. Unsupervised HMM - based Approach

Based on the strengths and limitations of the methods outlined above, we would recommend the following framework to detect suspicious survey responses that could prove to be invalid:

#	User ID	Rule-based Approach	LSTM	HMM
1	866	X		X
2	616	X	X	
3	1272	X		X
4	953	X	X	
5	365	X		
6	505		X	
7	727			X

Every time a user completes a survey, the generated data would run through the

three different models and each model would generate a "suspicion flag" depending on whether the user's response seemed suspicious and potentially invalid or not. Considering the limitations of the LSTM and HMM models due to lack of accurate data used for training the models, we would suggest putting the most emphasis on the "Rule-based Approach", which relies on solidified logical rules and was trained on 94% of the entire dataset. Therefore, if the rule-based approach flags a response, it is highly likely to be invalid. If either LSTM or HMM also flag that response, the probability that it is an invalid response goes up, thereby driving the response higher up the suspicious-flagged list. Responses that were flagged only by LSTM or HMM would still be considered suspicious based on the user's mouse paths, however, it would be up to the client to decide whether those are worth looking into - this decision would be entirely up to dotin depending on their goals and priorities.

Based on this framework, dotin can also experiment with using different thresholds for the rule-based approach. Currently, our logic is that if the user failed to pass either the score, reading time, or topic rules, we flagged him as suspicious. However, dotin can also choose to make the threshold stricter and only flag a user as suspicious if at least 2 of the 3 rules are failed, or even all three.

Business Implications

Identifying a machine learning methodology to determine the validity of online paid surveys is a new turf and no company nowadays is using such a technology. The implication of creating such a methodology can have a drastic impact on the cost for companies to collect and validate data. Not only such a method would prevent fraudsters from being paid, but more importantly, it would ensure that the data collected is correctly filled, which is far more valuable in the long run. Some estimate that *1 survey out of 5 contains fraudulent responses*, yet incentives for taking the survey might increase or decrease this estimate.

For dotin, the immediate business implication to using our methodology means that there is no wasted time to gather new data and a decrease dotin survey costs of at least 44% (only taking into consideration the results from the Rule-Based approach). After completing the survey, only the users whose

responses are correctly validated by our methodology will be paid, the others will be discarded, preventing fraudulent data to even enter the system.

Appendix

Figure 1: Table of Features Developed

Feature Category	Feature Name	Feature Description
Mouse Activity	Click Count	Number of times a user has answered a question
Mouse Activity	User Record Count	Number of times user has performed any mouse activity (scroll+moves+clicks)
Mouse Activity	Validation	Target Variable for supervised machine learning (boolean); classification modeling
Mouse Activity	average_click_delay	Average time taken between one click and the next; aggregated by user
Time	Max time lapsed	Total time taken by the user to complete the survey
Time	Time since last movement	Total time since the last mouse movement
Time	Time since last click	Total time since the last mouse click on a radio button
Time	Factor of difference	Quantify how the time it takes each user to complete the survey compares to expected read time calculations.
Distance	Total Distance	Total distance traveled by the user (Euclidean distance)
Distance	Measure_width_covered	A feature to give us a measure of screen coverage by user in terms of width (x coordinate)
Distance	Measure_height_covered	A feature to give us a measure of screen coverage by user in terms of height (y coordinate)
Direction	Moves left , Perc of left movements	The count and percentage of instances when the user moves from right to left on the screen
Direction	Moves right, perc of right movements	The count and percentage of instances when the user moves from left to right on the screen
Direction	Moves up, perc of up movements	The count and percentage of instances when the user moves from bottom to top on the screen
Direction	Moves down, perc of down movements	The count and percentage of instances when the user moves from top to bottom on the screen
Direction	No horizontal movement	Count and percentage of instances when user shows no horizontal movement on the screen
Direction	No vertical movement	Count and percentage of instances when user shows no vertical movement on the screen
Survey	Bf_votes_1,2,3,4,5, Bs_votes_1,2,3,4,5, Miq_votes_1,2,3,4,5, pgi_votes_1,2,3,4,5,6,7	Choice of answers for each category for each question
Survey	Bf_abs_min_max_response, Bs_abs_min_max_response, Miq_abs_min_max_response, pgi_abs_min_max_response	Checks whether the user has selected all 1s (absolute minimum value of question choice selection) or 5s/7s (absolute max value of question choice selection) per question category type (bf_questions, bs_questions, miq_questions, pgi_questions). Boolean
Survey	Standard deviation on similar questions	Checks how user responses deviate on questions that are similar in nature

References

Arapakis, I., Lalmas, M., & Valkanas, G. (2014). Understanding Within-Content Engagementthrough Pattern Analysis of Mouse Gestures. (pp. 1439-1448). Conference on Informationand Knowledge Management. https://dl.acm.org/citation.cfm?id=2661909

Churchill, E. F., & Navalpakkam, V. (2012). Mouse tracking: Measuring and predicting users'experience of web-based content. Conference on Human Factors in Computing Systems.https://dl.acm.org/citation.cfm?doid=2207676.2208705

Guo, Q., & Agichtein, E. (2010). Towards predicting web searcher gaze position from mousemovements. (pp. 3601-3606). CHI EA '10 CHI '10 Extended Abstracts on Human Factors inComputing Systems. https://dl.acm.org/citation.cfm?doid=1753846.1754025

Elbahi, A., Omri, M. N., Mahjoub, M. A., & Garrouch, K. (2016). Mouse Movement and Probabilistic Graphical Models Based E-Learning Activity Recognition Improvement Possibilistic Model. Arabian Journal for Science and Engineering, 41(8), 2847-2862. doi:10.1007/s13369-016-2025-6

Horwitz, R., Kreuter, F., & Conrad, F. (2017). Using Mouse Movements to Predict Web SurveyResponse Difficulty. Social Science Computer Review, 35(3), (pp. 388-405).https://doi.org/10.1177/0894439315626360

Stamp, M. (2017). A Revealing Introduction to Hidden Markov Models. Introduction to Machine Learning with Applications in Information Security, 7-35. doi:10.1201/9781315213262-2