



Sanjana Chowdhury's Writing Samples

Some cool things I've written about
business planning + strategy,
artificial intelligence,
and other things

December 2020

Table of Contents

About	1
AI Literature	
Prologue	2
Challenges in Deploying Explainable Machine Learning.....	3
Beyond Feature Importance: Explaining Uncertainty Estimates.....	14
Business Planning + Strategy	
Prologue	26
Technica Experience Director Guide 2019	27
Dash In Store #1208 Business Plan	37

Author's Note: use the linked bookmarks to take you to the desired sections

About

Hello! My name is Sanjana. I graduated from the University of Maryland, College Park in May 2020 with a BS in Computer Science and BA in Linguistics. I've done some pretty cool things in my time. I worked at NASA and helped introduce machine learning to the ExoMars rover. I built up two very successful collegiate hackathons over the course of 4 years. I worked at a startup and helped them build their AI community.

This is a collection of a few things I've written for miscellaneous reasons. The intention is to showcase my ability to develop strategy, communicate effectively, deliver quality content, and grow. I believe this collection is a small glimpse into what I can do and who I am.

This collection is divided into different sections. There is a prologue at the beginning of each section detailing the context of each entry - why it was written, brief background, a smidge of self-criticism, etc.

Please feel free to use the Table of Contents to navigate through the document. I suggest liberally using the links to take you to the parts that are of interest and watch the page numbers to keep track of where you are.

I hope you enjoy, and thank you for reading!

Sanjana

AI Literature: Prologue

From July 2020 to November 2020, I was a Technical Program Manager Intern at a startup called [WhyLabs](#). There, I helped run their AI community called [Rsqrd AI](#). Part of my responsibilities was to turn several tech talk recordings into blog posts.

I've written 8 blog posts for the community on topics ranging from using synthetic data to the importance of building reliable pipelines. My posts have received a lot of recognition and praise from AI leaders in the industry.

There were several challenges in making this happen.

1. The content of these talks were vastly different from anything I've learned about AI/ML before. Every talk was a different subject matter I had to learn quickly, and I had to get comfortable enough to write an entire article on the content I just learned.
2. I had no model or precedent on how to write these posts. I had to figure out on my own what the best way was on how to present the information in a sustainable, efficient manner.
3. I needed to determine how to best communicate the information to both technical and non-technical audiences - how can I make this easy to understand but still engaging for everyone?

The following two entries are blog posts I've written for the community under the field of explainable machine learning. The first one, *Challenges in Explainable Machine Learning*, was written in September and the second one, *Beyond Feature Importance: Explaining Uncertainty Estimates*, was written in October.

To write these posts, I had to quickly get familiar with the field of explainable ML. I spent quite some time understanding the research presented in both pieces to determine how to best present them.

Note the use of timestamps and the liberal amount of links to external resources throughout the posts. This was my solution to guide all kinds of readers through the article: if they wanted to go into further detail or didn't understand a topic, they could find a hyperlink for their needs. This also helped me write content quickly - if there was something I felt like I couldn't explain well or I thought would ruin the flow of the article, I would just send the reader to a timestamp in the recording or an external resource.

Challenges in Deploying Explainable Machine Learning (w/ Umang Bhatt)



In this talk, Rsqrd AI welcomes Umang Bhatt, Ph.D. student in the Machine Learning Group at the University of Cambridge and Research Fellow at Partnership on AI! Umang speaks about the challenges in ML explainability and his work in collaboration with Partnership on AI on ML explainability in the industry.

How are existing approaches to explainability used in practice? [Answer: only used by developers]

The following is the work of Umang and multiple collaborators and was also presented at the ACM FAccT 2019 Conference as the paper [Explainable Machine Learning in Deployment](#).

Umang mentions how there's been massive growth in the literature surrounding XAI (explainable AI). In particular, there has been an increase in proposed algorithms that aim to "explain" machine learning output. With this observation, one of the things Umang's team set out to do was study how organizations use these algorithms.

For their study, they had semi-structured interviews with 50 people from 30 different organizations, with interviews lasting between 30 minutes - 2 hours.

Shared Language

The team realized there needs to be shared and established language for thinking about explainable AI to be able to have these conversations around the topic.

Umang provides the following two definitions:

- Transparency: Providing stakeholders with relevant information about how the model works: this includes documentation of the training procedure, analysis of training data distribution, code releases, feature level explanations, etc.
- (Local) Explainability: Providing insights into a model's behavior for specific datapoint(s)

The key point is that explainability is an algorithmic approach to transparency.

Questions That Were Asked

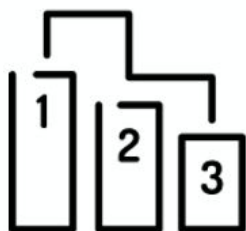
Once the shared language had been established in the interview, the interviewees were asked the following questions:

- What type of explanations have you used (e.g., feature-based, sample-based, counterfactual, or natural language)?
- Who is the audience for the model explanation (e.g., research scientists, product managers, domain experts, or users)?
- In what context have you deployed the explanations (e.g., informing the development process, informing human decision makers about the model, or informing the end user on how actions were taken based on the model's output)?

Types of Explanations

Umang and his team found certain types of explanations that were very popular:

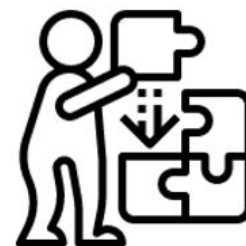
Types of Explanations



Feature Importance



Sample Importance



Counterfactuals

- Feature importance: using features of relevance (important) over less relevant features (unimportant) to make the right decisions and avoid misleading conclusions like spurious correlations
- Sample importance: attempts to answer the question “which training data points are the most influential when predicting on a new test point?”
- Counterfactuals: attempts to answer the question “what do you need to change about yourself in order to change your outcome with respect to this predictor?”

Who Cares about Explainability?

The next thing they realize is who are the stakeholders that have a vested interest on ML explainability:

Stakeholders



Executives



Engineers



End Users



Regulators

- Executives: the team found that corporations, engineering managers, chief data scientists, etc. had experiences where their executives would ask them to start using explainable ML so they can say that their company uses explainable ML
- Engineers: they are the most popular stakeholder. ML engineers and data scientists use these explainability techniques to sanity check their models.
- End users: developing consumable explanations creates trust between end users and AI
- Regulators: regulators, such as laws, diplomats, and the research community, have a vested interest in understanding how XAI works to see how it affects the surrounding community.

Final Findings

The team finalizes 3 key findings of the study.

Explainability is used for debugging internally

As seen earlier, explainable AI is used primarily by ML engineers for ML engineers. They kind of add this layer of XAI on top of their existing pipelines to check their models. These systems don't find their way to the end user as exclaimed by some XAI literature. The primary uses for these systems that they found were in content moderation and in finance.

Goals of explainability aren't defined internally

They found that important stakeholders are not involved from the beginning which ultimately makes it hard for explanations to be consumed by the right people. It's

important to engage with these stakeholders and define the purpose and role of these explanations and to establish explainability goals.

Establishing Explainability Goals



Technical limitations make explainability hard to deploy in real-time

There were some frequent limitations that came up in the study.

- Spurious correlations exposed by feature level explanations: if I find a problem with the feature attribution, how do I fix it?
- No causal underpinnings to the models themselves: execs and regulators want to throw things into a model and want to see its impact on the outcome: they expect to see some sort of correlation. The models might not be causal at all and support this system that creates correlations. From this stems causal explanations for non-causal models, and disinterest from execs and regulators because there isn't causal understanding.
- Sample importance is computationally infeasible to deploy at scale: which training points are similar to the model and create similar outcomes? That's hard to answer, especially at scale.
- Privacy concerns of model inversion exist: if I can give you explanations, I can invert your model and invert your training data and reconstruct it, and it becomes a huge privacy concern.

Can existing explainability tools be used to ensure model unfairness? [Answer: not in their current form]

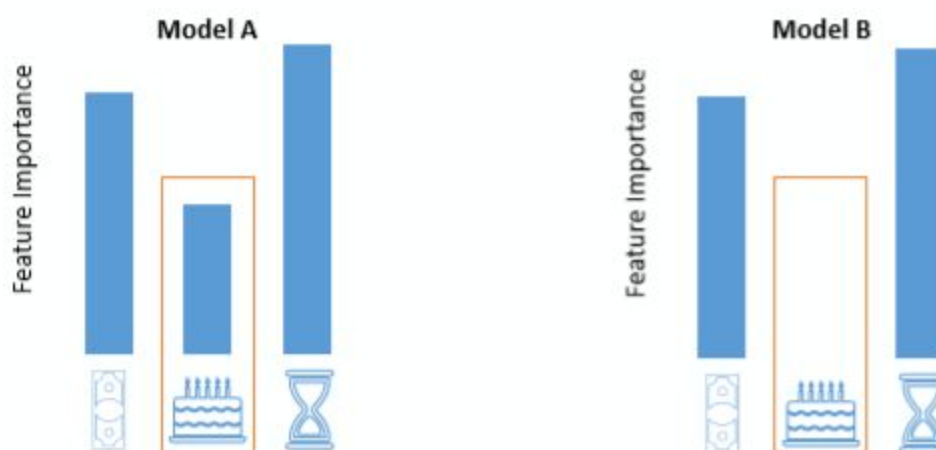
The work that Umang talks about that answers this question comes from another paper Umang worked on called [You Shouldn't Trust Me: Learning Models Which Conceal Unfairness From Multiple Explanation Methods](#).

The key (and slightly depressing) takeaway from this section that Umang emphasizes is:

Feature importance reveals nothing reliable about model fairness.

Why should people care about good explanations?

Heed the following example from Umang.



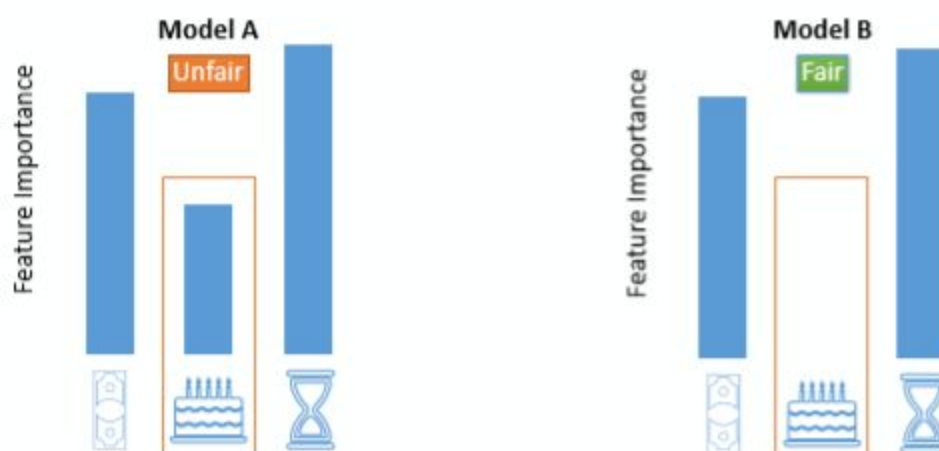
mock models with the same accuracy, but different importance assigned to different features

Here, there are two mock models with 3 features: 1. wealth 2. age, and 3. time spent on doing some downstream tasks. Say someone uses the same explanation techniques on them.

The explanation on the first model ranks the importance of the features as time > wealth > age.

The explanation on the second model ranks the importance of the features as time > wealth, and age is unimportant.

These models can perform the same with the same accuracy, but their explanations are vastly different. In theory, a regulator can come in and say that model A is unfair because it uses a discriminatory attribute, age.



model B is deemed more fair than model A because of the importance of 'age'

This problem begs the question: how can someone turn model A into model B without losing accuracy?

Attempting to manipulate explanations

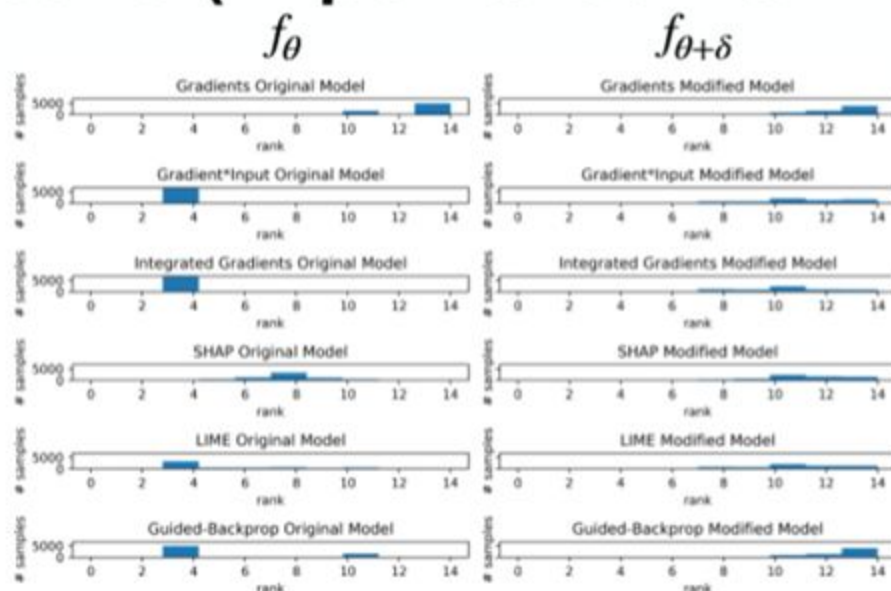
There is some work already that's gone into manipulating explanations. The solution explored in this paper is downgrading explanations with adversarial perturbations on the parameters such that the authors can hide the unfair or discriminatory feature. Or simply explained, attacking the parameters of a model such that the attribution given to a certain feature (like age) decreases.

Results

The setup and method used in the paper can be found at [25:50](#) in more detail. The goal of the experiment parameterized by θ is to find some δ that gives the aforementioned results. Here, f_{θ} is model A, and $f_{\theta+\delta}$ is model B.

What Umang and the authors are looking at is importance rankings. Here, they are looking at 14 features. They can take feature importance rankings and just rank features based on the magnitude of attribution given to them. The paper reports the magnitude and ranking of the discriminatory features with respect to different explanation techniques.

Results (Importance Ranking)



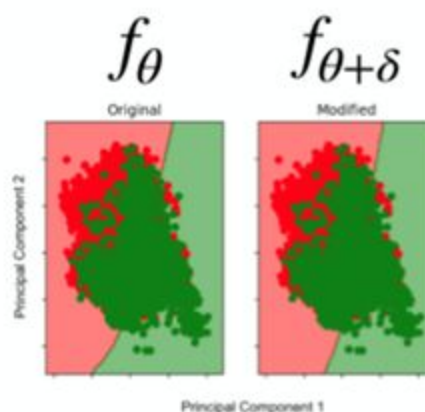
importance rankings of a discriminatory feature in model A (f_{θ}) and model B ($f_{\theta+\delta}$)

The thought behind this is that a regulator can look at the rankings for a discriminatory feature. If the ranking is very high in an explanation such as in model A, the regulator can say that the discriminatory feature is being used too much and that another model would be better. In model B, the authors show that they can flatten out the gradients with respect to multiple explanation techniques and scale back the magnitude of attribution given to a feature.

TLDR; the paper's adversarial explanation attack does 3 key things: significantly decreases relative importance, generalizes to test points, and transfers across explanation methods, along with the following key findings:

Findings

- Little change in accuracy, but difference in outputs is detectable
- Low attribution achieved with respect to **multiple** explanation methods
- High unfairness across multiple fairness metrics (compared to holding feature constant)



How can we create explainability tools for external stakeholders? [Answer: community engagement and thoughtful deployment]

This work is called [Machine Learning Explainability for External Stakeholders](#), and it was based on a closed convening hosted by the Centre for the Future Intelligence, Partnership on AI, and IBM.

The Discussion

In this activity, 33 participants from 5 countries came together in New York in early February.

They were 15 ML experts, 3 designers, 6 legal experts, and 9 policymakers. They came from a wide variety of domain expertise with backgrounds in finance, media, healthcare, and social services. The goal was to facilitate an inter-stakeholder conversation around explainable machine learning.

There are two main takeaways:

1. There is a need for community engagement in the development of explainable machine learning.
2. There are nuances in the deployment of explainable machine learning.

Umang proposes to tackle these takeaways as questions to be answered by the community and further research.

Community Engagement

The following questions should be asked when engaging with the community.

1. In which context will this explanation be used? Does the context change the properties of the explanations we expose?
2. How should the explanation be evaluated? Both quantitatively and qualitatively
3. Can we prevent data misuse and preferential treatment by involving affected groups in the development process?
4. Can we educate external stakeholders (and data scientists) regarding the functionalities and limitations of explainable machine learning?

Deploying Explainability

In the same way when considering community engagement, the following important questions should be asked when deploying explainability.

1. How does uncertainty in the model and introduced by the (potentially approximate) explanation technique affect the resulting explanations?
2. How can stakeholders interact with the resulting explanations? Can explanations be a conduit for interacting with the model?
3. How, if at all, will stakeholder behavior change as a result of the explanation shown?
4. Over time, how will the explanation technique adapt to changes in stakeholder behavior?

All of these things are completely context dependent, and that's why it's important to have community engagement from creating the model to iterating on it.

Conclusion

It's become critical to be able to explain machine learning and to be able to answer questions on where outputs come from. There's increasing interest from stakeholders in explainable AI, but it's a lot easier said than done. While current explainability tools have a long way to go and are not sufficient in their current form, AI practitioners can continue to create meaningful explainability tools for stakeholders through community engagement and thoughtful deployment.

Cool Stuff to Check Out

[Explainable Machine Learning in Deployment.](#)

[You Shouldn't Trust Me: Learning Models Which Conceal Unfairness From Multiple Explanation Methods.](#)

[Machine Learning Explainability for External Stakeholders](#)

Interesting questions from the video:

- How strongly do regulators insist on explainable models? Are they open to non-explainable models? And if so, what are the desiderata? [20m 50s](#)
- Is there an agreed method to explain a given model regardless of where it's deployed? [32m 30s](#)
- What are your thoughts on liability, policy, and safety standards in spaces like self-driving cars in regards to explainable ML? [38m 34s](#)
- What are the overall incentives of using explainable AI? [55m 43s](#)

Video: [Rsqr AI - Umang Bhatt - Challenges in Deploying Explainable Machine Learning](#)

Slides: [Challenges in Deploying Explainable Machine Learning](#)

All information and ideas presented in this post are that of the speaker and the talk.

Beyond Feature Importance: Explaining Uncertainty Estimates (w/ Javier Antorán)

Uncertainty in ML



In this talk, Rsqrd AI welcomes Javier Antorán, PhD student at the University of Cambridge! Javier talks about the existing approaches to interpretability and how a new approach using counterfactuals can be used to explain uncertainty.

What is Interpretability?

Machine Learning interpretability is a broad term which is used loosely, but generally refers to the ability to explain or present something in understandable terms to a human. Someone could say an ML model is interpretable if humans are able to understand the reasoning behind its decisions. Others might go a step further and argue that a model is interpretable when humans are able to reproduce its reasoning and thus predict the model's outputs given its inputs (this is known as forward stimulability). [A really good overview of ML interpretability is given by Finale Doshi-Velez and Been Kim.](#)

Existing Approaches to Interpretability

Interpretable Data Driven Decision Making

There are many existing interpretable “data-driven-decision-making” models, like linear regression or decision trees. In these models, it’s very simple to find the reasoning behind a decision. A linear regression’s output is the sum of the weights of its input features, meaning features that have more weight directly have a bigger impact on the output. A decision tree is dictated by yes-no questions and once all the questions have been asked, predictions are made and are constant within the leaf node. It’s fairly simple to trace back a decision up the tree.

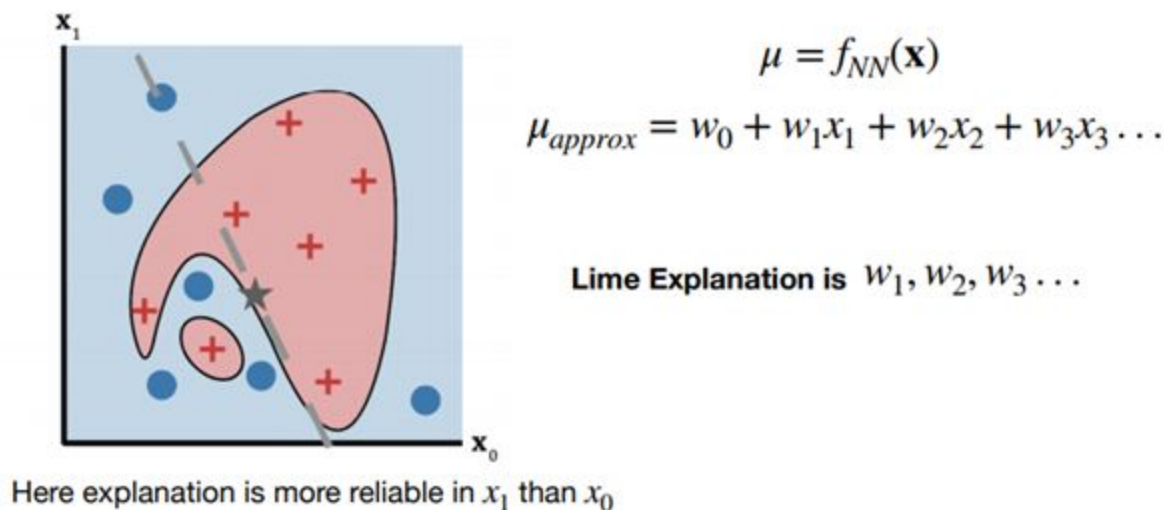
Not Very Interpretable Data Driven Decision Making

If one sacrifices interpretability and the ability to easily determine the pretense of an output (ability to correlate where a decision comes from), a wider range of predictive models become available, like neural networks. They can do some cool things like take in the image of a road and output driving commands. Given this, it’s not as easy to predict the decision making for the output of a neural network like linear regression and decision trees.

Feature Importance

LIME

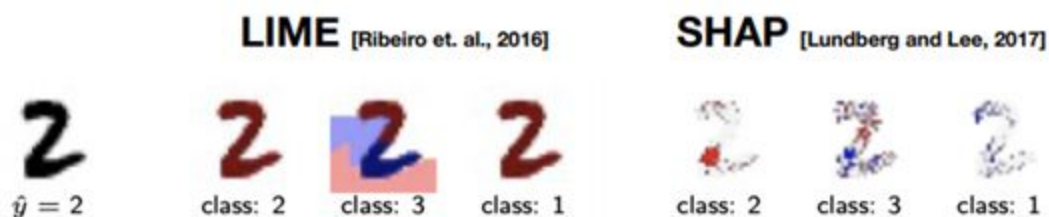
Once interpretability is sacrificed in applications such as neural networks, how does someone get it back? The most commonly used approach is feature importance, and within the family of feature importance techniques is [LIME](#) (Local Interpretable Model-agnostic Explanations). The idea behind LIME is that someone can approximate their nonlinear model with an interpretable linear function. For a more in-depth look at how LIME works, you can go to [5m 05s](#).



Applying LIME on a non-linear function

Feature Importance on Images

When approximating with linear models on non-linear functions, it can become meaningless on strongly non-linear functions. Javier goes into detail on an example where LIME falsely classifies the pixels of an image of the number '2' as positive evidence for the number '1'. The example can be found at [6m 52s](#).



Approximating becomes meaningless on extreme non-linear functions

Counterfactual Explanations

The alternative to feature importance is counterfactual explanations. The idea behind counterfactuals is that they capture the notion of how things would have worked out if some factor would've been different – the “what-ifs” if a factor were to change. An example that Javier gives is in the medical field when giving a medical diagnosis. Once a final outcome and diagnosis has been resolved from running several tests and procedures, it's commonplace to ask how an outcome would have been different had the medical staff done something differently. Another example of a counterfactual would be to think about “what if fish could fly? How would our

ecosystem be different?”. [Here’s an article that goes further in depth on counterfactuals.](#)

The interpretability community has started using counterfactuals to ask questions like what sort of interventions does someone need in order to get a desirable result – what would need to change? In the context of classifiers, it’s questions like:

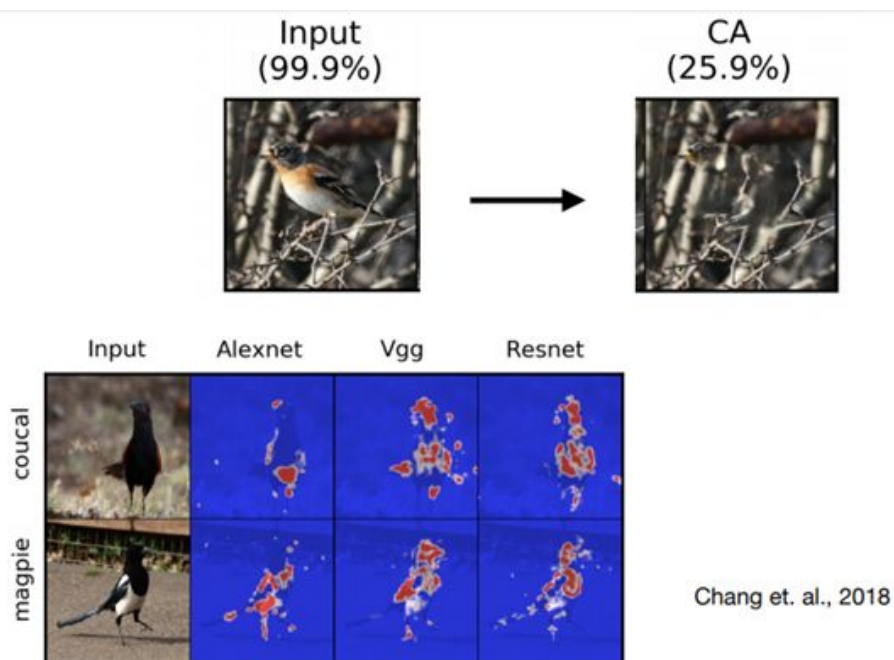
- What features would I need to remove such that my model’s confidence decreases?
- What features would I need to remove such that my model’s prediction changes?

The cool thing about counterfactuals is that they’re model-agnostic – their explanations have completely clear meanings with no need to approximate like in linear explanations in LIME.

Counterfactuals in Image Classification

Javier gives an example of counterfactual explanations in image classification.

Given a picture of a bird, each pixel can be removed and replaced. With each step, there’s a question that is asked: is it still a picture of a bird? This process can keep going until the model doesn’t think there’s a bird in the picture anymore, creating a collection of pixels that identified the animal. Full explanation can be found at [10m09s](#).



Chang et. al., 2018

Identifying the "bird" pixel by pixel

Uncertainty in ML

Despite neural networks having gained a lot of popularity recently, a well-known issue is that neural networks are very overconfident and tend to behave erratically in scenarios that are different from the ones that they've been trained for

As mentioned, neural networks can seem very confused and place high confidence in decisions a human wouldn't make. The solution to this is to build systems that are uncertainty aware - systems that can recognize where they could be confused.

Sources of Uncertainty

There are 2 reasons why a model can be uncertain:

1. There's noise in the data ([Aleatoric Uncertainty](#))
2. Asking about points very different from the ones in the training set ([Epistemic Uncertainty](#))

Javier goes further in explaining what uncertainty looks like in a detailed example at [12m 53s](#).

Most existing ML models have ways of expressing noise uncertainty: the probabilities outputted by a regular NN's softmax activation layer represent noise uncertainty. However, it is much more difficult to capture epistemic or "model uncertainty". This latter form of uncertainty captures the notion that the data that the model has seen during training was insufficient for it to make a confident prediction about the test point with which it is being queried.

In practice, the more complex the model, the harder it is to capture epistemic uncertainty. Neural networks are very complicated which makes it nearly impossible to quantify what these models “don’t know”. Instead ML practitioners resort to approximations, like [MC dropout](#). There is often a tradeoff between using simpler approximations which are more unreliable but scale to big networks and using more exact approximations that provide better uncertainty estimates.

Uncertainty in Practice

Here are some applications of how uncertainty is used in practice today:

- Robustness: no one would want self-driving cars or medical diagnoses to behave erratically when faced with something unfamiliar (One way to handle a situation like this would be to reject anything unfamiliar and ask for the intervention of a human driver or doctor, respectively)
- Dataset building, safety, and awareness: if there’s a model with uncertainty awareness, it can identify uncertainty in certain groups of data and reevaluate potential biases.
- Active learning: this is where there aren’t many labels but getting them is really expensive. In these situations, we want to identify the points which will be most useful to train our model. It turns out that the most telling points are the points with high estimates of epistemic uncertainty.

To play around with these ideas and implement your own Bayesian neural network, you can go to the [GitHub repo here](#).

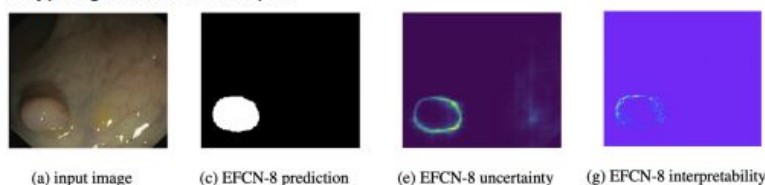
Are Models that are Aware of Uncertainty Interpretable?

Uncertainty-aware models are interpretable as regular ML models. If using linear regression as mentioned before, the model will be very interpretable. On the other hand, if using something like a Bayesian neural network, it probably won’t be interpretable.

However, uncertainty can also be used as a tool to help users understand their models’ predictions. Javier gives an example of a neural network identifying polyps in an image found at [19m 50s](#). Here, the Bayesian NN’s uncertainty around the detected region is more telling than the output from an interpretability technique (LIME).

- **Uncertainty can help users understand prediction in some cases**

Polyp segmentation example:

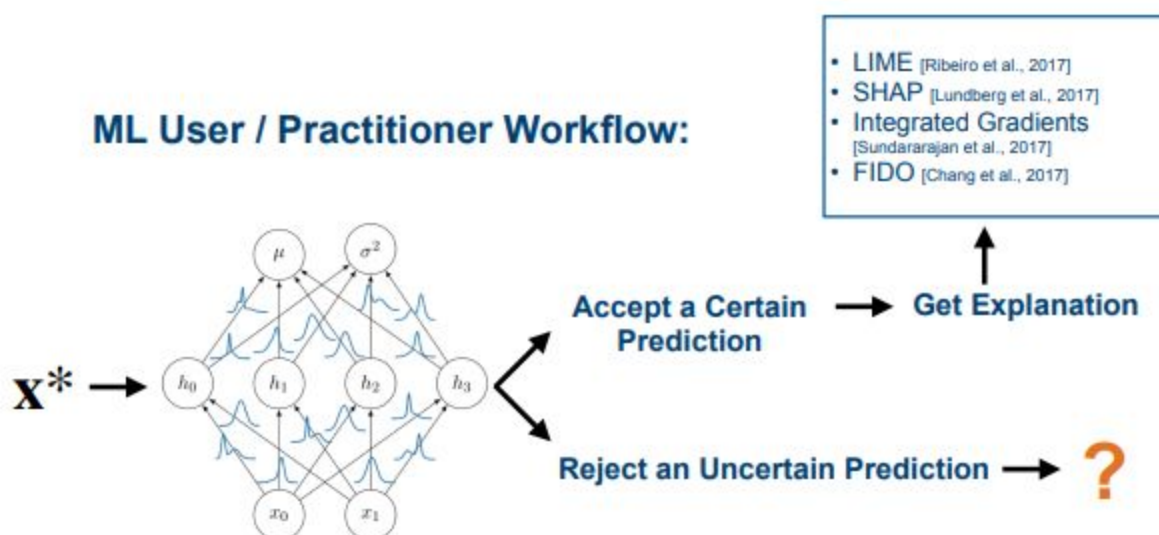


Wickström, et. al., 2019

Identifying polyps

What Happens when a Model Doesn't Know the Answer?

Until now, we have just considered interpreting ML predictions where our model is confident. However, interpretability may be needed the most when this is not the case.

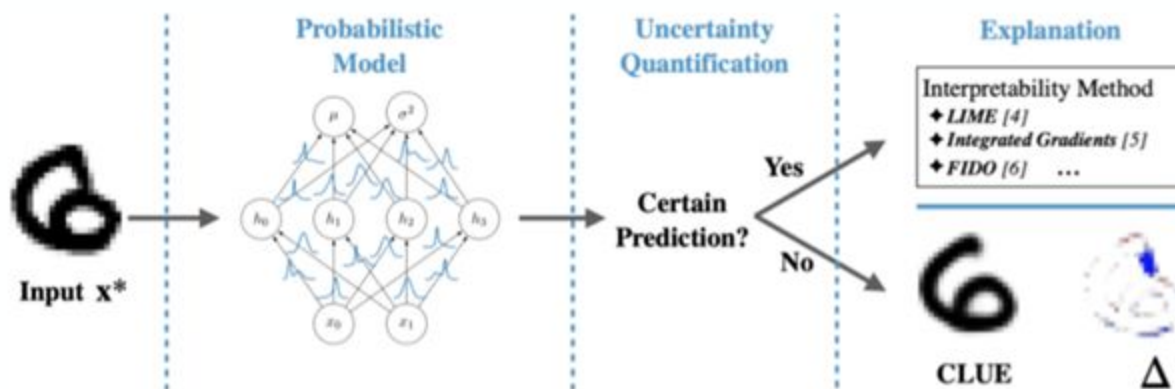


If a model decides to reject an option and doesn't know what to predict: what happens next?

CLUE: Counterfactual Latent Uncertainty Explanations

[CLUE](#) is a method that answers the question “what is the smallest change we need to make to an input, while staying in-distribution, such that our model produces more certain predictions?”.

How CLUE would be used in practice is given by the following example:



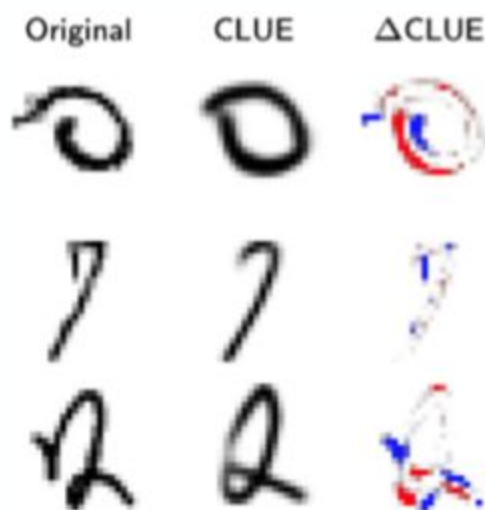
Given an uncertain image (here the ambiguous image could be a '6' or '8'), the reject option can be triggered. Then, CLUE would run and give the nearest certain image. In this example, CLUE takes out a line in the original image and now identifies the image as a '6' with confidence, highlighting what the issue was.



Javier goes further into explaining the algorithm at [27 m.](#)

Presenting CLUE to Users

CLUE is able to show the difference in the original input and its own prediction. Basically, what changed in the original input to make it look like CLUE's output. In the case of handwritten digits, CLUE shows which strokes can be erased to create a more certain prediction.



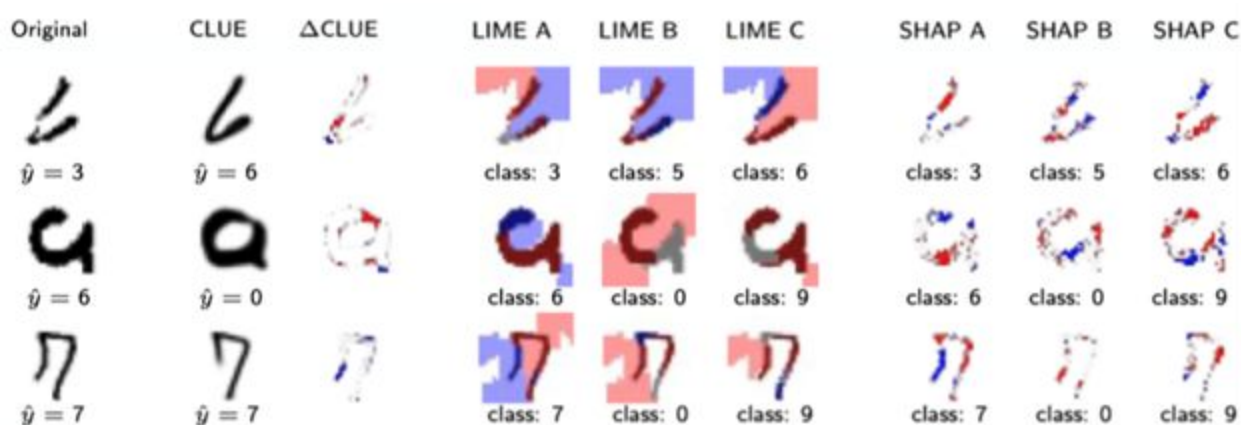
(a) MNIST

CLUE can also be applied to classification, regression, and image data.

CLUE vs Feature Importance (LIME/SHAP)

CLUE's counterfactual nature allows it to add more information than it starts with from the original input. CLUE can point out pieces (like different strokes in drawn numbers) that can be helpful or harmful to prediction. This is in contrast with feature importance techniques, like LIME or SHAP, which simply highlight different features if they provide evidence for or against a class, incapable of adding new information.

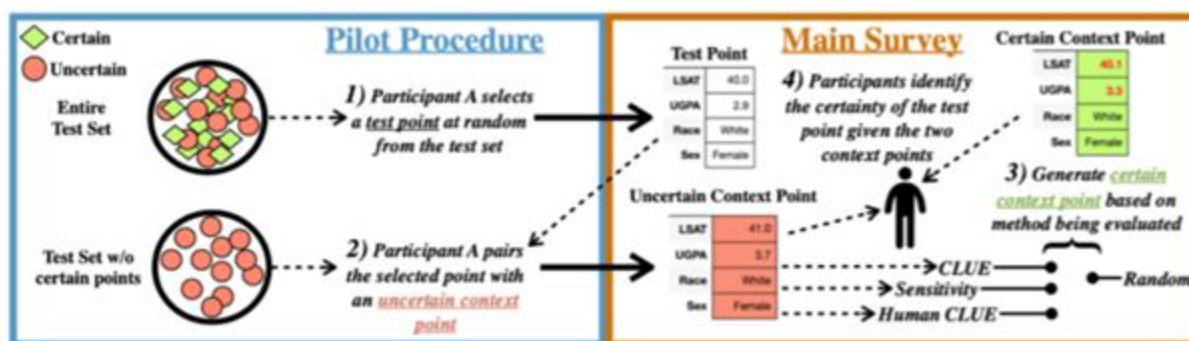
Javier goes over a great example where each of the 3 methods are used to interpret a model's predictions for some uncertain inputs. Here, traditional approaches to interpretability struggle due to a lack of strong evidence for any class or conflicting evidence for multiple classes. A counterfactual approach allows CLUE to give the most useful observations at [29m 41s](#).



User Study to Show the Value of Uncertainty Estimates

The authors of CLUE wanted to test and see if any of this information on uncertainty is useful and actually make a difference to practitioners. A human simulability task was set up with the idea that an explanation should give a user enough insight to be able to predict how a model is going to behave in an unforeseen circumstance.

The setup can be found at [31m 10s](#). There are 4 approaches (groups) tested: CLUE, Random (randomly select points), Sensitivity (similar to CLUE but doesn't take the step of restricting the hypothesis space of explanations to plausible inputs), and Human (actual people selecting relevant counterfactual explanations). Each approach has 10 "users" that will be shown its explanations. Each group is given some uncertain inputs and their explanations, and then asked to predict if their model will be certain or uncertain on some new inputs.



The results show that users provided with CLUEs did significantly better, with an accuracy of 82.22% over the average accuracy from the other approaches of 58.89%.

User Study: Results

Method	N. participants	Accuracy (%)
Random	10	61.67
Sensitivity	10	52.78
Human	10	62.22
CLUE	10	82.22

CLUE's improvement over all other approaches is statistically significant

(Using Nemenyi test for average ranks across test questions)

A similar experiment was held using images and can be found at [35m 36s](#).

Conclusion

Feature importance methods such as LIME have been valuable in interpretability, but as demonstrated, has its drawbacks such as an inability to represent extremely non-linear functions. An alternative to feature importance methods are counterfactual explanations that aim to tell how a decision was to change if some input factor were different.

Recently, the field of uncertainty aware NNs has gained a lot of traction. Uncertainty awareness opens up a whole new family of questions relating to ML interpretability which traditional methods are not well suited to answer. CLUE represents a first step towards reconciling ML interpretability and uncertainty. This method has shown success in helping users identify points for which their model is uncertainty and shows great potential in the growing field of interpretability.

Cool Stuff to Check Out

Paper: [Getting a CLUE: A Method for Explaining Uncertainty Estimates](#)
[Javier's GitHub repo on Bayesian Neural Networks](#)
[Towards A Rigorous Science of Interpretable Machine Learning](#) by Finale Doshi-Velez and Been Kim

[What's a Counterfactual?](#)

Interesting questions from the video:

- Can you talk a little bit about how CLUE compares to other explainability techniques and what gives it an advantage over others? [38m 38s](#)
- Could counterfactuals be used to understand the effect of missing data and maybe optimal replacement missing data? [45m 16s](#)
- Can you speak a little more on determining distribution data on CLUE? [50m 03s](#)

Video: [Beyond Feature Importance: Explaining Uncertainty Estimates](#)

Slides: [Rsqr AI - ML Interpretability: Beyond Feature Importance](#)

All information and ideas presented in this post are that of the speaker and the talk.

Business Planning + Strategy: Prologue

For different reasons, I've had to prepare many strategies, plans, and guides on how to do things. This first started when I became an Experience Director for [Technica](#), the world's largest all-women's hackathon, in 2018 and I had to plan an entire 8 months of work to get ready for the event.

This was by no means easy. The Experience team was new that year and there wasn't much content yet to guide me through the planning and what I should be doing. The team was only given a few goals and the rest was up to interpretation.

Through 2018 and 2019, I was able to give definition to the role and make it successful. It was no longer a vague position with an ambiguous purpose. After my time as Experience Director was over, I made it my mission to document everything I've done to the best of my ability for the next director. This is the *Experience Director Manual 2019*. It details what the next director needs to know, my experiences and advice, and how I've done things. My hope was that it becomes a reliable document for the next director to answer any questions they might have and give them direction - something I had to find on my own.

The next entry, *Business Plan for Dash In Store #1208*, is something I wrote for my dad. My dad bought a gas station convenience store in early December 2020. He had to submit a business plan to receive franchise rights. My dad owns a liquor store and already knows how to run a business, but he didn't know how to create a plan or where to even start.

I wrote his plan and developed his strategy. Using a sample plan (which, by the way, was very poorly written) given to him by his accountant, I interviewed my dad and got some information like his past and some plans he has for the store. From there, I developed a mission, set standards, and created a timeline for the store. Stringing everything together, I hoped to create a brief and informative plan that's easy to read.

I'm not saying the plan is perfect or even any good. I don't have complete knowledge on how to run a convenience store. My goal was to create a plan that was satisfactory enough to gain franchise rights. To my surprise, my plan received a lot of praise when getting approval, and my dad was very impressed.

Experience Director Manual 2019

By: Sanjana Chowdhury

General Info

Role

The experience director is in charge of all things experience, which include both organizer experience and hacker experience. As the experience director, you are a leading force in creating Technica's atmosphere. It's your job to make an active effort to know everyone on the team, be active on Slack, establish relationships with everyone, and make sure everyone at Technica and the organizing team are having a good time. Your goal is to have the team want to come back next year. Think about it like being the fun big sister that has mom's credit card. You want to engage with everyone and just create a really good experience.

History

Jordan Mess and Sanjana Chowdhury were co-directors in 2018, and Sanjana was the director again in 2019. If you have any questions, you can reach out to her.

Hacker Experience

Hacker experience is the experience involving day-of Technica. This includes swag, mini events, sponsor experience, and logistical stuff.

Swag

This is probably the biggest job on the team. Swag includes:

- Hacker shirts

- Hacker swag
- Stickers
- Lanyards
- Pins
- Organizer shirts
- Organizer swag

Quotes are posted in the Technica folder. There are a lot of vendors who reach out. In 2019, we worked with Freshprints. The POC was Esther Zhang who was conveniently on the sponsorship team. Reach out to her again to establish a new relationship for 2020. In addition, a lot of other vendors also reached out to the experience email, but they reached out too late. It's worth investigating ahead of time with each vendor and see what quotes you get.

Shirts

Technica is known for their high quality shirts. I would recommend sticking to the same type of shirt for both hacker and organizer shirts. It's worth getting samples way ahead of time to determine which shirt you want. Here are some of the quotes for the shirts:

- Hacker and volunteer shirts: 5.25/shirt
- Organizer shirt: 11.75/shirt

Heather is the more expensive material, but it's completely worth it. Make sure you really compare quotes and find vendors that can match or beat these quotes.

For shirt sizes, get more mediums than you think you would need. It's the most popular size. Also, make sure you have enough XL and XXL sizes. These are a lot more in demand than you'd think.

For organizer shirts, we usually have a gradient somewhere in the design and have printing in two places. Sleeves are popular.

Hacker Swag

Technica always gives out signature swag in addition to the shirts, usually on Saturday night after dinner. In 2017, we gave out water bottles. In 2018, we gave out blankets. In 2019, we gave out hats. The goal is to keep the cost of these signature swag \$5 and under per item. You should order more than you'd think you need for organizers, sponsors, mentors and volunteers, and for promotion. In 2019, we ordered 900 hats and got through almost all of them.

Technica gives out swag for people who demo. In 2019, we gave out little glitter rocket lamps that were 4.75/lamp. Like signature swag, you should aim for \$5/item or less. In 2019, 600 lamps were ordered and we went through almost all of them.

Stickers and Pins

Stickers and pins are both available from Stickermule, and you can get a discount code from MLH. For stickers, we usually have a signature "Technica" logo sticker for that year, two variations of sticker sheets, and the rest is for funsies and what the design team comes up with. In 2019, we got decals and big clear stickers that say "I Technican". Feel free to have fun with it and work with design to come up with some cool stuff.

Technica has pronoun pins, usually "they/their", "him/he", and "her/she". The designs are already made from back in 2018. In addition to pronoun pins, you can do themed pins. In 2019, experience had themed pins with little planets on them. They are very cute. These pins should all be 1 inch in diameter.

Lanyards

Lanyards determine the difference between volunteers, mentors, organizers, sponsors, and hackers. You should have 5 different colors. In 2019, the width of the lanyards was 5/8inch. These were a cute size, but you can size up to 3/4inch and get thicker lanyards like in 2018 and 2017. Make sure you get the little badge holders for the lanyards so people can put in a name tag.

Organizer Swag

Each year, we tend to give wearable swag for organizers. In 2018, they were softshell jackets. In 2019, they were Patagonia fleeces. In 2018, the jackets were \$36/jacket. In 2019, the fleeces were \$96/fleece. You need to discuss options with your executive directors and see what costs they would be okay with. Organizer swag is usually a surprise from everyone, including the directors. Only the experience, executive, and design directors should know about the surprise swag.

In 2019, we also bought Technica shot glasses for the organizing team. It's always cute to get something small in addition to the big swag. These were also a surprise to the majority of the team. We also got sherpa blankets for select advisors, best volunteer, and people we really think helped out and deserved it. They were like \$40/blanket. These were handed out sparingly.

It's important to do thorough research when choosing which organizer swag to get. This means reaching out to different vendors and really getting the lowest price. Look at Amazon reviews, company websites, etc.

*****Swag is REALLY time consuming and hard. The earlier you start and the more you delegate, the easier it will be to handle. Don't wait until the last minute to order something or to work with design because you need at least 2 weeks shipping time for large orders and for peace of mind*****

Mini Events

Mini events are the little fun side events at the hackathon. There's usually 2 done on Saturday night. In 2019, we had 'make your own' succulents, 'make your own' galaxy jars, a nail station, and a spicy ramen challenge. These were all really fun! But here are some do's and don't's for mini events:

DO:

- Have a mini event where everyone can participate
- Have mini events where people can make something and take it home

- If having a messy mini event, make sure the surrounding area has tarp all around
- Have mini events be no more than 2 hours long
- Make sure you have enough supplies for the time of the mini event
- Have enough volunteers

DON'T:

- Have a food related event. Children are brutal and messy.
- Have fragile materials like glass. Too much of a liability
- Have super messy events that risk staining and damaging the armory
- Have events that take way too long or are a lot more work than it's worth, like pictionary

It's super important that you coordinate your mini events ahead of time with other events, such as workshops and dinner. It's hard to do mini events like dancing. There simply isn't enough space in the hacking venue. There's also no space for mini events to happen in the rooms of the armory because they are used for either workshops or sleeping. Make sure your mini events can fit around the sides of the hacking space.

In 2020, you should definitely aim for different mini events. Succulents have been done for 2 years, and change would be nice. You should also aim to stay under budget. \$2K is plenty for mini events.

Logistics

Logistics involves things that are day-of and are more logistical. This includes registration, team formation, the photobooth, toiletries, and judging. These are tasks that just need to get done.

For team formation, there should be slides from previous years to use an example. It's a pretty effective way to do team formation and it's gotten a lot of compliments. It might be good to do 2 rounds of team formation. One at the beginning of the hackathon, and another one a little later for people who come in late or still don't know what to do.

Remember to put up the photobooth and ask design to make props, and to also put up the banners for the other areas such as snack and registration.

Toiletries need to be bought and restocked every so often in the bathrooms. There's a lot left in storage. Go through inventory and figure out what you need.

Judging is coordinated with the tech team through the expo app. Get as many judges as possible, at least 20. There's always so many demos, and you can't risk not being on time. Reach out to organizers and mentors for help with Technica prize categories.

- Technica has a few prize categories. One is related to that year's theme, and the other is related to beginner for high school and beginner in general
- When picking a winner, it's important to share the wealth. The winning team should not have won another prize from a sponsor.

Sponsor Experience

Technica aims to provide a great experience for sponsors. This means really trying to make sponsors happy at the hackathon. It can get incredibly boring and tedious just sitting at a booth. In 2019, Technica made sponsors bags for each table filled with fun goodies, and also went around and gave each sponsor a hat. It's a little bit of Technica for them to take home that leaves a lasting impression on the sponsor for them to want to come back next year.

In 2018, experience did sponsor bingo. In 2019, we gave special buttons for sponsors to give out. There's always been trouble with trying to engage hackers with sponsors more once hacking starts. Maybe you could come up with a fun game, or have a separate event for sponsors.

Other

Passport System

In 2019, experience worked with workshops to create a passport system. For each workshop attended, hackers could earn a stamp. They could then trade in these stamps for prizes at different tiers. This was a way to encourage hackers to go to workshops and make sponsors happy by having impressive attendance at the workshops they paid for.

This was wildly successful. Attendance was the best it's ever been and hackers were eager to get all three tiers of prizes. They engaged with the event a lot, with the average number of workshops attended being 3. This is definitely something that Technica should do again in 2020, but with more preparation ahead of time.

Here's how the passport system worked. Everyone got a mini spiral notebook (ordered from Amazon) when they registered. At the end of each workshop, someone from the workshops team stamped the notebook. They would then come to the help desk and get their stamps marked off, and trade them in for swag in the following tiers:

- 2 stamps: a magnet
- 3 stamps: a sleeping mask
- 5 stamps: a mug

In 2020, there is a lot of potential to expand the system. Maybe have more tiers? Maybe give out stamps at more events? It is worth to take some time and really flesh out the system.

Best Volunteer

In Technica 2019, experience worked with event ops to pick the best volunteer. This was just a private channel in the day-of Slack where organizers can vouch for a volunteer who was working super hard and thinks they should deserve the award. This was a small side gig, but it was successful. The best volunteer(s) was recognized

at closing ceremony and were shipped sherpa blankets. They really liked it a lot, and it's a nice way to show volunteers they were appreciated.

This system can be expanded. It can be expanded to mentors, or give out smaller prizes to more volunteers. This is something worth investigating and shows that Technica appreciates the support they get.

Organizer Experience

A big role of the experience director is creating a good organizer experience. This is making the organizers feel appreciated, making them feel like a part of the Techni-fam, and making them feel like organizing Technica is worth their time.

Bonding Events

Experience is developed a lot through bonding events. It's imperative that there are bonding events throughout the organizing season such as grind nights, team bbqs, and movie nights for organizers to get a chance to engage with the team and also show how Technica appreciates them.

Since a lot of the organizing season is during the summer, bonding events as a team can't be done during then, but you can set up Technica meetups during the season for whoever is in the area. For example, you can have a Seattle, New York, and DMV meetup.

When school starts, you should have a bonding event at least twice a month.

Technica would be 2 months away at that point, and you should try to find ways to engage the team as much as possible. Some ideas are:

- Bonfire
- Game nights
- Going out to DC
- Grind nights

It's hard to find a space for so many people, so some good places are startup shell and Oakland's MPR room.

Team dates are also good. It's a smaller group of people so finding something to do won't be as hard. The point of team dates is to encourage people and teams who don't normally interact with each other a chance to do so. The exp director would assign the teams and timeframe, and the directors would be responsible for setting up the date. It's a good, hands-free way for experience to encourage team bonding.

A popular place to do team dates are at the spot mini.

It's super important to have a solid team bonding timeline for the year right after you pick your team. It takes a lot of planning and commitment, and for these to be successful, you need to set the tone for the team from the beginning.

Director Culture

A team's culture all starts with the director. It all trickles down. If you have execs that are close, the directors are inclined to be close. If the directors are close, then their teams are inclined to be close, too.

It's important to have director bonding events like potlucks and happy hours to encourage directors to engage with each other. It's also important for the team to see that so they can get a feel for how comfortable they should be with the team. It also encourages the director to get closer to their own team which makes happy and appreciated organizers.

The directors don't have to be best friends, but they should be comfortable around each other and set an example for their team, especially new members. You should encourage directors to bond with their team and create a welcoming community within their realm.

You should sit down with the directors and ask them what Technica means to them and things like that. It will help you understand the tone of the team and give you a starting point to see what you'd might like to change, improve on, or maintain.

Timeline

March

- Pick your team

April/May

- First meeting
- Bonding event with your own team

June/July

- Have meetings and brainstorm swag ideas
- Create bonding event timeline

August

- Establish relationships with vendors and find best quotes
- Establish relationship with design team to see if they have any preliminary ideas

September

- Finalize swag ideas
- Work with design to get swag designs started EARLY
- Figure out mini event ideas

October

- First and second week: finalize and order swag ASAP
- Finalize mini events
- Figure out sponsor bags
- Order all the supplies you need for sponsor bags, mini events, and anything else by the last week of October

November

- Figure out logistic stuff(team formation, judging)
- Tie up any loose ends

Advice

While it can be overwhelming and emotionally exhausting, being the experience director is the most fun job on the team. Your job is to literally make sure everyone is having a good time. Just remember to relax, smile, trust your team, and to trust yourself. If you're having fun being the director, then you're doing it right.



Business Plan for Dash In Store #1208

Monir Chowdhury

[Email]

December 2020

Dash In Store #1208, 261 South Dupont Highway, New Castle DE 19720

Executive Summary

Dash In store #1208 has significant potential to generate steady revenue that has piqued the interest of current business owner Monir Chowdhury. Under Monir's management, he plans to smoothen employee operations, fix store infrastructure, and entice passersby to come in and find something they like, no matter how menial the purchase.

Mission

Dash In #1208's mission is to have something for everyone and not have anyone leave empty handed. Whether it's a bag of chips or a toothbrush, Dash In #1208 wants everyone to find something they needed or discover something they didn't think they wanted before.

Core Principles

There are several core principles that are key to a successful operation

1. Be meticulous about keeping track of sales and merchandise
2. Keep a steady flow of communication between stakeholders
3. Treat everyone with respect
4. Maintain a clean and safe environment
5. Keep a sharp eye for any risks endangering the store and solve any problems right away

Objective

The goal for the first year is to get settled and invest heavily into the store for any modifications or repairs. After this first year, net gain is expected to continue growing every year thereafter at 5%.

Owner Qualifications

Monir Chowdhury is a seasoned business owner who is looking for investments where he sees long-term growth and stability.

Since 2005, Monir has been involved in managing, growing, and working in small businesses. After working at a 7/11, he became the owner of a liquor store called Stock Market Bar (Hanover, MD) and has been running it for the last 15 years. In 2019, he sold a bar called Limelight Inn (Catonsville, MD) after it ran under his ownership for 8 years.

Monir knows what it takes to run a store. From things like checking and buying inventory, managing employees, dealing with customers, and even fixing a roof – Monir has been on the front lines of running entire operations and continues to do so relentlessly with success. From his experiences in both current and past ventures, he understands what it takes to run a business, assess risk, deal with failure, and how to handle difficult situations.

Monir plans on using his retail management and entrepreneurial experience in small business to make Dash In a sustainable and profitable store. He will be leading all management.

Description of Business

Company Ownership

The franchise will be owned by an LLC created by Monir that will be named later. All measures are in place to cover anticipated costs such as franchise rights, repairs, and inventory.

Location

The location of this store is 261 South Dupont Highway, New Castle DE 19720 right off Pulaski Highway. The space is shared with an Exxon gas station. The area in general is bustling with retail traffic with most notably Christiana Mall a short 5 minutes away. There are also other shopping centers such as a Walmart nearby as well.

Hours of Operation

As of right now, the hours of operation are from 6am to 10pm every day. This will continue as new ownership takes charge, but Monir will be experimenting with staying open as late as 12am and see if the extra 2 hours will be valuable in bringing in more profit.

Products and Merchandise

Monir intends to keep and maintain all current products and merchandise and will take a closer look at what more products he can bring into Dash In after taking ownership. He intends to introduce fresh food and for it to become key in bringing in more patrons and increase total sales.

Suppliers

Monir will use the store's current suppliers as he gathers information on which suppliers offers the best deals and products will help Dash In #1208 succeed the most.

Management

Monir will be the main person behind running the operation. As mentioned before, he currently has another business and is a seasoned business manager. He will oversee things such as payroll, accounts, work schedule, etc. like he already does at his other business. After overhauling operations, he will select a few personnel and train them to help him manage the store.

Competition

Dash In will mainly be taking advantage of Exxon's traffic. We need to take into consideration other gas stations in the area. There is a Wawa and Shell gas station further up the highway, and a bp and Valero on the other end of the highway as well. Furthermore, there are Shell gas stations closer to Christiana Mall.

To make Dash In #1208 competitive and bring in more patrons, it needs to keep up with the options available at the Wawa. This means bringing in hot food and making sure Dash In delivers quality customer service.

Conclusion

The goal for Dash In #1208 is to be a reliable convenience store. It plans to build off the former owner's establishment and expand on the current offerings of the store. In addition, Monir wants to improve the quality of the service and space that will ensure a welcoming environment to customers. He believes if the store can provide a safe and clean space where people are free to roam around and find what they want or need, he can make Dash In successful.