**CRAIG:** Hi, I'm Craig Smith and this is Eye on AI.

**CRAIG:** This week we're talking about no-code machine learning, a subject that I’m particularly interested in.

**CRAIG:** Technology often follows a familiar progression. First, it’s used by a small core of scientists, then the user base expands to engineers who can navigate technical nuance and jargon until finally it’s made user-friendly enough that almost anyone can use it.

**CRAIG:** Right now, the process for building machine-learning software is making that final leap. Just as the clickable icons of Windows and Mac OS replaced obscure DOS commands, new “no-code” platforms replace programming languages with simple drag and drop interfaces. The implications are huge: Where it used to require a team of engineers to build a piece of AI software, now users with a web browser and an idea have the power to bring that idea to life themselves.

**CRAIG:** In this episode, Jonathan Reilly, co-founder of Akkio, a no-code AI platform, takes us on a tour of the technology today and what it holds for the future.

**CRAIG:** Before we begin, I want to mention our sponsor, ClearML, an open-source MLOps solution. If you are building models without a no-code platform, try ClearML. You can give them a spin at clear.ml.

**CRAIG:** Now, I hope you find the conversation with Jon as informative as I did.

**CRAIG:** Hey John. If you can introduce yourself, give a little bit of background, and then what Akkio is, and what you're doing, and then I'll start asking questions.

**JOHN:** Okay. My name is John Riley. I'm one of the co-founders of Akkio which is a no code machine learning platform. we founded Akkio specifically to make it easier for people who are not technical to start to take advantage of AI and machine learning and leverage it to understand the patterns in their data that are driving their key outcomes.

**JOHN:** And then use models to make real time data driven decisions because we believe. In the next several years, almost every process that's data driven, which is almost every process in every business, will be made more efficient and enhanced by the use of machine learning to optimize those processes. And so, we're working really hard to build a machine learning platform that makes it incredibly easy for anyone who has access to data to start to leverage ML.

**CRAIG:** I've used, Akkio a little bit myself. I wrote a piece on betting on the horses with Akkio and I want to get back to that because I think it was working. I lost access to the data, but the primary models and the primary use is classification of tabular data.

**CRAIG:** Is that right?

**JOHN:** We actually have three different model types right now and, I'd say classification is probably about a third of the use. It was definitely the one that we built first. so, there’s three core model types.

**JOHN:** One, which is your outcome is in a category and that could be like positive or negative, or good or bad, or could even be like a customer support outcome that has, 15 different possible resolutions and you're taking some text and trying to figure out which resolution would be the right answer for your user.

**JOHN:** But there's two other types that are worth mentioning, one of which is regression. That's where you're trying to predict a numerical outcome. And that could be, what's the lifetime value of a certain customer,

**JOHN:** or it could be you're buying a property in a market, and you've got some historic market data and you'd like to estimate the sales price of that property given a bunch of information about it, like the number of rooms and the square footage and its quality and stuff like that. Numerical predictions are also possible,

**JOHN:** but then more interesting is the ability to leverage time as an input variable. So, forecasting or time series modeling, is the third type. And about 50% of our customers have a lot of interest in the time driven analysis. Because if you think about it, like one of the hardest things to really account for in any given environment is the changes that happen with time.

**JOHN:** It's a really hard thing to model. And machine learning can actually learn the patterns with respect to time and forward forecast revenue or practically anything that you have measured over time. And so, we see more and more models that leverage that time component as well.

**CRAIG:** Yeah, I spoke quite a bit with one of your engineers who was developing the time series model. How is it performing now? Because we were looking at stock prices and there are so many variables that go into pricing of a stock, most of which are unpredictable, that it wasn't as clear cut as I had hoped. Are there certain cases where time series make sense and others where it doesn't?

**JOHN:** Machine learning can really only learn the patterns in historic data to predict what's going to happen in the future. An example, I like to use is hurricane tracking, which has some decay, window of accuracy associated with it.

**JOHN:** If you've ever seen one of those hurricane cones, they're reasonably certain where it's going to be in a day or two and a lot less certain where it's going to be in a couple of weeks. It's very much the same thing with time series forecasting. And so, we try and show in our models how much of your accuracy decays over time.

**JOHN:** But there's a second piece there that goes into forecasting those hurricane tracks, which is a lot of historic data as well as current data around how pressure systems and other environmental changes like water temperature and temperatures where the hurricane's going to pass through, are going to impact its strength, its speed, all of those features, right? The same thing applies to that stock problem, and it is actually quite complicated because there are some historic patterns that go into driving a stock price.

**JOHN:** The stock price yesterday is probably a pretty good predictor of what the stock price will be tomorrow most of the time, just like the weather yesterday is a pretty good predictor of what the weather will be tomorrow. But of course, the weather is impacted by seasons and stock values are impacted by market dynamics.

**JOHN:** And so, if you really want to start to think about predicting something like a stock price accurately, not only do you need to know the historic price of that stock. But you need to know all of the driving factors associated with the price of that stock. And putting that data set together is really complicated because there's so many different moving pieces. and that's why you see these specialized hedge funds and quant firms that are able to actually put the time and effort into gathering the necessary data to start to make better forecasts about what's going to happen with stocks.

**JOHN:** If we had all of that data, I like to joke, we would be at quant fund and not building an ML platform for everybody. so, we do get a lot of customers who come in looking to predict things like stock prices and you can do it to an extent, but I wouldn't rely on it for investing because it's very unlikely that an individual stepping into that space is going to have access to the underlying data necessary to do it accurately.

**JOHN:** If you remember when we were doing the horse racing examples, there is a lot of data associated even with predicting the outcome of a race, which is like that horse's relative performance against other horses, the turf type, historical races, the length, it goes on and on, right?

**JOHN:** And the people doing a good job or accurately, leveraging machine learning in those very specific areas where monetization really matters, the advantage is often gleaned through the quality of the underpinning data that you have. And those markets are very efficient, right? Like in, in a betting market, people are using their brains to pour over the data and try and understand it, and the experts are setting the line.

**JOHN:** So, you have a consensus understanding of what's happening driving that price. It's similar with stocks too, like your stock price is in a way sort of a collective output of everybody thinking about all the things they know and deciding what should be fair for the price of that stock. And you can get an advantage, but it's pretty data intense,

**JOHN:** so, so I wouldn't recommend those types of applications, but a business trying to forecast their revenue is a much, much simpler problem than predicting the price of the stock because, you have all of the information you need associated with that. Your funnel, what your deal pipeline looks like, how it's worked historically or historic seasonality, you typically have all of that information right on hand. So that's where you start to be able to leverage time series inside of a business in a more effective way.

**CRAIG:** That's fascinating. Can you tell me a little bit about how you put the models together?

**CRAIG:** There's a lot of these models in libraries and how you connect it to the platform, how the platform selects a particular model, because that's a dynamic process, as I recall.

**CRAIG:** Also, whether you are following the research and either tweaking the models or swapping in new models or improved models or developing other modules with models that are emerging?

**JOHN:** Yeah. So, I'll start by saying it's our CEO, Abe, who primarily writes the underlying ML engine.

**JOHN:** We automate the selection of the best model type for your problem. There's a lot of different approaches to machine learning modeling, different neural architecture structures, different types of models.

**JOHN:** we use auto ml, and it basically tries out all the different types of models or a bunch of different types of models, sometimes combinations of types of models, sometimes stacked models where one model does something and then feeds its output into another model and it tries to find the model that performs very best for a given data set.

**JOHN:** And it tries to do that in a really fair and unbiased fashion and so to accomplish that, you take 80% of the data available at training time and you bootstrap the modeling exercise, testing and validating. And we use a thing called cross entropy loss to pick the winning model.

**JOHN:** It is a fair way of finding the best performing model relative to other model performance. And once you pick that best performing model against the 80% of the data, you then apply the 20% that you held back randomly and didn't look at to the model that you've trained.

**JOHN:** And then you see how well it did against data that it's never seen before. And you generate a bunch of data patterns and insights through that process. Now that whole process is pretty complex. And in fact, historically, a data scientist would try different model types on their own and tweak different parameters and come up with the best performing model that they could make.

**JOHN:** What Auto ML does, is it automates that process and makes it so that you don't have to go through all of those steps. And then, because our users are not data scientists and they're less technical, we abstract that away under the cover of a user interface where you can simply connect your data and hit train with a target output variable in mind, and we will automate all of that model selection for you and come back with a report that says, Here's how well we were able to detect patterns in your data.

**JOHN:** Here are the patterns in your data that are impacting your outcome and how much they're impacting it. And then you can deploy that model in a couple of clicks in the cloud, and you can run inference or predictions against it for future outcomes to make real time data driven decisions. So, you can leverage it in a couple of ways.

**JOHN:** One is strategic decision making. Looking at the patterns in history that are relevant to your outcomes and how much they matter to decide, where to invest or spend your time, but also if you have real time data flows that need to be sorted or processed or organized, you can do that

**JOHN:** too.

**JOHN:** And we are always adding new model types into that engine to compete for which one's the best performing model. So, I think we're up to something on the order of 25 different possible basic architectures and then combinations of them scale from there.

**JOHN:** But we do stay abreast at the latest ML research. And make sure that we're using all of the latest model types. And that's important because the space is growing really fast right now and there's lots of improvements in efficiency and approach that allow you to get to the best answer faster than you used to in the past, and with less compute cost.

**JOHN:** And that helps the user not have to sit around and wait for results for a day or something. You can often have your answer in a matter of minutes now, which is great.

**CRAIG:** Yeah, the whole no code movement is fascinating to me because I don't code, and it is growing quickly. A lot of the big AI companies are putting money into no code platforms themselves.

**CRAIG:** I've spoken to Microsoft about their power app platforms. How do you compete in that market, given that there's some very big players?

**JOHN:** We compete specifically by being fast and efficient at the process of picking your best model and then being really easy to use. And that's like a two-front investment from us in ease of use and insight generation coupled with deep tech. And because we're a startup, we're able to optimize around user experience and benefits in a faster way than many large companies are because they have lots of different constraints in place that guide their behavior or their investment philosophy.

**JOHN:** Many other auto ML platforms on the market, even from the very big auto ML vendors, are built under their cloud compute teams. And their cloud compute teams are not really incentivized to reduce the cost of compute because they get paid when you use cloud compute.

**JOHN:** And there's some incentive structures there that have allowed us, to be 60 times faster at delivering a good result, with the same exact performance and, some technical lift on our side to accomplish those tasks too. But really the other piece is many of the early players in the space are building data science tools for data scientists and very technically competent users. That's what happens anytime technology emerges; the first set of users are incredibly competent and very technical. We're drawing a hard line and focusing more on a less technical user, and that's where the no code term comes into play.

**JOHN:** If you've tried to use some of these other platforms on the market, you'll find no code is maybe a stretch. They sometimes use the term low code. You might be working on a command prompt to accomplish some of your tasks. That's unnecessary in our platform. It means saying no to some things.

**JOHN:** There's lots of knobs or levers that data scientists would like to turn that we don't surface in our platform, but we do that because we have to stay very focused on ease of use and we're working on a set of customers who aren't going to know how to set those knobs or dials anyways so we try and automate the setting of them to the best possible setting.

**JOHN:** In general, though, I think no code is a really fascinating emerging space. I like to think of it as layers of abstraction, of instructing computers, what to do. Even people who write software today aren't writing machine language, like a low-level code, right? They're compiling that from a programming language, and in a lot of the emerging no code space, you're using either a graphical user interface or sometimes natural language to generate the software code, which is then compiled to run on a machine.

**JOHN:** So, you're just getting increasing layers of abstraction. But with that comes the ability for more and more people to start to leverage a technology to good effect across a larger range of applications. And so that, I think that's what's most exciting and it's actually enabled in many cases by machine learning.

**JOHN:** We have a new feature that we haven't shipped yet, but we're showing around in some demos where we can actually transform a data set by natural language. So, if you wanted to take a data set and write a command, remove all personally identifiable information from this data set, you could just write that command in natural language and a machine learning engine would convert it and then it would look through the dataset and it would determine automatically what's likely to be personally identifiable information and not.

**JOHN:** And it would remove it and then it would give you a preview of the results showing you what it removed, and you could just hit confirm. And that'd be a really fast, efficient way to accomplish that task. The old way of doing that task would be, to do a bunch of specified filters or really in-depth action to try and get it out of every possible column.

**JOHN:** It might be in every possible row, it might be in, that'd take a lot of time. Or you could, write some code that hunted it down and tried to remove it from the data set. But instead with a simple command, you can start to make this stuff happen. And. It takes you from pseudo code things like long, if this, then that statement's in Excel, like I'm sure many of us have written and turns them into natural language statements.

**JOHN:** Make a column that contains yes, if this is true in the record, otherwise put no in there and you can just state it in natural language, and it'll apply it to the entire dataset in the snap of your fingers. the same thing's happening with writing software, mL enabled software tools can help make writing code faster and easier and more efficient.

**JOHN:** So I think you'll see that the space is going to undergo a really rapid change where you're able to give a more natural language explanation of what you'd like to accomplish in the past, that would take someone like me, a product person talking to an engineer, asking them to build something for me, and in the future I can perhaps have that conversation with an interface and it will build something for me and it'll do it really fast and efficiently. So that gets exciting I think, for all of us who don't have the technical depth. Now, the trick or the rub of it is, you need to be quite good at really specifying the constraints of the system.

**JOHN:** a lot of the engineering expertise is not just being able to understand a coding language, but understand corner cases and what do you do when a record is not clear? And I think the trick will be learning to articulate what you want with clarity, even covering corner cases or building feedback loops in the no code platforms that allow the user to see, okay I've under specified for this case, I need to put a little bit more effort into describing what I want to have happen there.

**JOHN:** And, if you've ever worked with software engineers, that happens naturally. You ask for something and they come back and say, you didn't really think this through. How about this? Then you go back and forth for a while. So that'll happen too with the platforms, but what's exciting is you can do anything from standing up a website to working in databases and applying machine learning to inputs and, taking data driven output decisions without needing to code.

**CRAIG:** I've written a bit about the auto generation of code, and I've spoken to Deep Mind about Alpha Code and people behind something called TiCoder, the computer has a conversation with the user before it generates code to make the users intent clearer. Those are all based on large language models. How do you guys implement that natural language front end. Are you drawing on a large language model or is it something simpler?

**JOHN:** For several areas of our platform, particularly for that AI driven data prep feature where you can just ask it a natural language question, we do something probably fairly similar to if you were trying to write software code with a natural language query, which is, we try to use the natural language version of your query to interpret the transformations that you're trying to apply to that data set.

**JOHN:** And then we convert those into a program that applies those transformations to the data set and surfaces them back to you so you can see if it did what you want. And those are interpreting your asks by leveraging large language type models. And of course, large language models can vary by degree of specificity of expertise, right?

**JOHN:** So, writing sequel queries and stuff like that is a certain area where you can have the models that are reasonably good at performing that type of task versus other models that generate open ended text responses to queries or requests. So, everyone in the space is standing on the shoulders of giants in a way because of the underpinning technology. I think we're all going to be shocked at the speed of really rapid change that's going to happen across almost every digital data driven touchpoint you could imagine, from generated images to generated video, to text, to making business decisions with data.

**JOHN:** And it's all starting to really accelerate as the capability of the underlying tech advances. But yeah, large language models are very helpful in a lot of different application spaces.

**CRAIG:** It costs a lot of money to build and train a large language model? Are there models out there that you can tap into through an API or something for that function?

**JOHN:** Yeah, several companies make their large language models available via API. Costs are actually dropping exponentially right now, and those models are quickly becoming commoditized. It's our view that they'll be commoditized within a couple of years.

**JOHN:** Of course, new tech will keep pushing the state of the art further forward but those are expensive to do once but after done easier to leverage and use. And so, the answer there is I think the accessibility of a large language model to more and more companies to use in their products or in specific applications is going up and to the right exponentially.

**JOHN:** And I think people will be able to host them themselves very quickly as well and train them. The state of advancement there is happening pretty rapidly. And the thing I'd point to is just for the cloud providers of large language models calls against them and costs are dropping.

**CRAIG:** What are some of the more interesting customer applications that you've seen? And I wanted to talk about how Sterling Strategies is leveraging Akkio in the political fundraising space. can you talk about Sterling and then any other equally intriguing applications that you've come across?

**JOHN:** Sure. We have a pretty broad range of applications because we're building a platform for machine learning and data analysis. Sterling is a good example of a political fundraising company that helps candidates raise money from donors who are aligned with their views. And the challenge there, if you're trying to raise money for a candidate is basically half of the country might be on the same side of the political spectrum as a given candidate.

**JOHN:** And some small portion of them would be willing to donate to the cause, and most of them wouldn't. And so, you've got a giant sorting problem, which is how do you find the people who are most likely to donate to a given candidate considering what you know about their past, their past donations, their past political alignment, perhaps how they voted, if that's in the public record.

**JOHN:** And the answer is that's just a classification problem or a lead scoring problem really. And machine learning's really good at it. And so, they're able to take a database of political donors. And for each candidate score the probability that one of those past donors will donate to any given candidate.

**JOHN:** And then the outreach mechanism is pretty spread out. But they have a call center. They call people and ask them to give money. I'm sure you get political emails now as we're going into November. And by sorting and focusing on the people who are more likely or have a higher propensity to donate, which the ML model helps them see, they more than doubled the effectiveness of their outbound calling in terms of raising money.

**JOHN:** And of course, that's a big win, right? And that's the same type of win you'd get in a business if your if you have a sales team and you're calling leads to try and sell them a product, if you can focus your efforts on a subset of your pipeline that is most likely to convert, your ability to convert them goes up, your efficiency and converting them goes up.

**JOHN:** And if conversely, you can focus your efforts away from the ones least likely to convert, you save a bunch of money and time, your entire operation gets more efficient. Most businesses are trying to find the people who are a best fit for your product as opposed to not a fit at all.

**JOHN:** But we have customers using us in a pretty broad range of applications from a big freight forwarder forecasting arrival times and reasons that things are stuck in customs costs of shipments. Stuff like that, given the massive data of history they have about shipping between ports. We have insurance companies predicting things like costs or back to work times given an injury type. We have customer support teams classifying records so that they can route them to the right person to, to handle like an open-ended text input.

**JOHN:** Someone comes in and says, hey, I'm having a problem with your product. This is the problem in plain text described. You can build a model that'll get that request routed to the right person to handle it and maybe even automate a suggested response back to them immediately, so they don't have to wait around specific to your business.

**JOHN:** This is not a general model, right? It'll learn from your past resolutions on these issues, and it goes from there, teams who are doing like data driven decision making on their marketing spend. Where do they invest and where do they not and which types of leads Are they investing in in real time?

**JOHN:** It's pretty broad. We have some companies doing even HR stuff where they try and identify employees that might be at risk of leaving the company so that they can intervene early, and they look at past patterns of employees leaving and figure out like what the indicators of departure might be.

**JOHN:** So that they can go save their good employees. Just a pretty broad range of possible applications. Effectively, anywhere where you might try and look at some data analysis in a business and make some better decisions, you can start to leverage ML even to horse racing betting and we have people predicting crypto prices and stuff like that too.

**CRAIG:** Your platform does all the heavy lifting once the data is prepared. And as you were saying that data prep is oftentimes complicated. What formats can you ingest data in and are there limits to how much data you can handle?

**JOHN:** Yeah,

**JOHN:** there's some practical limits to any large data problem.

**JOHN:** We'll take data in directly from your database if you're using like Big Query or Snowflake or something like that, or directly from a CRM like Salesforce or HubSpot. But we'll also take CSVs Excel sheets and JSON files and other formats. The interesting thing is the practical limit in an Excel file is like a million rows, and we'll do 10, a hundred million row data sets.

**JOHN:** We've seen pretty big files. The practical limit starts to be with total size, which is like how many columns you have, how much of it's text, our rough rule of thumb right now is under 10 gigabytes on the training side, and you should be reasonably good to go. Over that and we might have to do some work on scaling.

**JOHN:** Now that covers almost every use case though because there's this thing about machine learning, where it's learning the patterns in your historic data, right? And if you show it a hundred examples, that might not be enough for it to learn patterns, depending how complex those patterns might be. The example I use typically with customers is, if you were doing photo recognition and you were just trying to classify a bunch of pictures as pictures of a dog or pictures of a house, that'd be a pretty easy problem because there's not a lot of overlap between a dog and a house.

**JOHN:** So, you might not need that many example pictures to show the model before it learned what the differences were. But if you're trying to show that the difference between a dog and a cat. There's a lot more intersect, and you need a lot more pictures. And if you're trying to teach you the difference between a dog and a wolf, you're going to need even more pictures because the similarities get even closer.

**JOHN:** The same thing applies in tabular business data. If the classes are cleanly separated, you can get away with a few examples. If they're pretty fuzzy and there's a lot of Venn diagram intersect between them, you're going to need more examples. But at some point, showing it another picture of a dog or a cat isn't really going to drive more performance.

**JOHN:** It's learned all the differences it can, and now you're just wasting time and compute resources at excess spend. So, at that scale you're going to get pretty much the best pattern recognition that the ML model can actually extract from the data. If you have like larger problems than that you're going to start using data science tools for incredibly technical people just like you would anytime you get into a deep tech problem.

**JOHN:** So, we're not really so focused on covering that crazy, super big model, with super important accuracy because most businesses we think in the next, five or 10 years, are going to have dedicated data science teams that handle those types of projects. So, we're covering the 90% of cases that you deal with in your everyday business, where an off the shelf solution is going to get you there and it's going to get you there faster and easier than anything else you can do with a group of users who frankly couldn't do it any other way.

**CRAIG:** You were using the example of computer vision there. Does Akkio classify images?

**JOHN:** No.

**JOHN:** No, we can't. We only deal with tabular data.

**JOHN:** We don't do video or images or audio files or anything like that. I just use that as an example that I think helps people like imagine the class differences and similarities in a better way.

**CRAIG:** On the tabular data, you were talking about how many rows, but on the columns, is there a sweet spot that for how many parameters or how many columns in an Excel spreadsheet you can handle?

**JOHN:** Today, to make the results return in reasonable period of time. We've drawn an imaginary line at 250 columns as the upper limit of columns that the model will attempt to figure out like the relative patterns contributing between, that's arbitrarily set.

**JOHN:** And on a go forward basis, as compute capacity increases like it always does, that'll expand as well, as well as be able to like, deliver results in a really quick period of time. Something like 250 different possible features or columns of input into your data that the model can learn from and figure out how to weigh in terms of what's driving your outcomes.

**CRAIG:** Yeah. Because talking to Sterling Strategies, they were saying that on each voter they can have up to 500 columns of data, everything from what TV shows someone watches to, whether or not they own a dog and that sort of thing.

**JOHN:** We try to intelligently select the 250 columns we end up using if you feed us more.

**JOHN:** And of course, like a user themselves can determine which ones they think are relevant if they want and exclude the others. So, there's a couple approaches there. Like with anything, there's some practical limits, which we've set in a place where almost never come into play. And by the time you're dealing with 500 possible drivers of an outcome, almost certainly there's 10 that are meaningful, another a hundred that like give very tiny contributions to your outcome.

**JOHN:** And the remainder extraneous information that's not particularly relevant. And we try and suss that out so that we obviously make the best prediction we possibly can within some time constraints primarily. Not spend days trying to get you an answer, but to allow you to iterate pretty quickly.

**JOHN:** And a lot of the time, that iteration loop is very critical to getting to an answer because often what happens is you might include data that happens after your prediction target in time, and then you have to go back and remove it. Causally linked data where like one of your inputs is exactly defining the thing that you're interested in predicting.

**JOHN:** And you need to remove that input from your prediction. Otherwise, the model will be a hundred percent accurate, but you don't need machine learning for it. When you find out you make that mistake, removing that input column and retraining wants to be a fast exercise, not another weight, several days type of exercise.

**JOHN:** And so, to the extent that we're drawing limits in terms of total file size or columns that you can throw into the platform we're doing so with the user experience and fluidity of results in mind. And we're trying to draw them in a way that the vast majority of anyone we've ever worked with, never comes close to hitting.

**JOHN:** And for people that do really need that 251st column, we could turn you on with a larger column space on our side, but you probably want to have a data scientist working on the problem if that last column, that last mile is so important to you. And we're not trying to replace data scientists, we're trying to enable Excel users to start to take advantage of machine learning too.

**CRAIG:** And as I recall, the platform identifies which columns or which features are contributing to the prediction and which are not or contributing less.

**CRAIG:** Is that right?

**JOHN:** That's right, yeah. One of the things we realized early on, is machine learning can often be a bit of a black box, and you could tell how accurately it's predicting a given outcome by back testing it or by throwing some data it hasn't seen against it where you know the outcome and comparing how well it did to the actual results.

**JOHN:** But for the non-technical user, that's not enough to really build trust in the system, right? Because they want to understand, and I want to understand how that machine learning model is actually making its decisions. Which inputs it's weighing in, its decision making and how it's weighing them. And so, we've put a lot of effort into developing insights that express the patterns in your data in a really easy to understand way.

**JOHN:** So, when you train a model in the Akkio platform, yeah, we show you how accurate it was, but we also show you how much it densified if it's a category problem, the outcome of interest. So, let's say you had a problem where you had 10% of your examples in the training set had purchased a product and 90% had not, or 10% had donated.

**JOHN:** And 90% and not, then you train the model and in that withheld 20% of data. The real question is if the model thinks you're going to be a yes or a donation, what percent of the time are those actually donations? And that's the densification of class. And so, we show you like, hey, when the model thinks that this person's going to donate, they donate 70% of the time.

**JOHN:** And that's where you get the business value, right? Yes. The model might not be a hundred percent accurate in saying, this person will donate, and they will always donate. But if you call the people the model thinks will donate. More than one out of two calls are going to convert.

**JOHN:** And if you ignore that people, the model thinks won't donate, you're not going to drop much business at all. So, you just got massively more efficient. And of course, there's some intersection there in the middle where your cats and your dogs look very similar or there's some uncertainty in the model of the outcome.

**JOHN:** And then you can figure out how to follow up on those two. So, we show you a densification factor, which is the business leveraging your model and then we show you each one of the columns, how much it's contributing to the predictive accuracy of the model and what the information inside that column means in terms of probability of outcome.

**JOHN:** So, if this were a housing price that we were trying to predict, it might be the home square footage is one of the number one drivers of price. Or it might be the city that you're in. If it's square footage, we'll say for low square footages, it's impact on prices minus this much for high square footages, it's impact on prices, plus this much over the average.

**JOHN:** Or for like locations, if you live in this neighborhood, if the house is in this neighborhood, that's a minus. If it's in this neighborhood, it's a plus. Because neighborhoods have like different house costs associated with them, and we'll show you all the driving factors. We'll even show you extracted value from text.

**JOHN:** These types of words in your text are associated with these outcomes. These types of words are associated with the other outcomes, and that helps the user understand the patterns and the data. Really cleanly, which they can make strategic decisions off of, but also understand how the model's making its decisions so they can feel more confident when they're leveraging it in real time decision making in their business.

**CRAIG:** Yeah.

**CRAIG:** One of the things that attracted me is Akkio's pricing model, and I used the freebie version. At what point does someone have to convert to a paid version. Is there a volume limit \ either on the data or the number of predictions?

**JOHN:** Yeah, so we have an open platform like you point out where anyone can come in and try it.

**JOHN:** If you make an account, you get a two-week free trial with all of the features enabled. There are some enterprise features we hold back from that trial because they're expensive on a compute basis. But if you talk to us, we'll set you up with an enterprise version of the trial as well.

**JOHN:** After the trial ends, we want people to still be able to train models and use them to some limited extent so that they can still learn about and use machine learning. So, we do have a free tier. It's not really marketed, but it does exist, and you can do, I think it's up to a thousand predictions a month, and you can train models in there and see their results.

**JOHN:** We don't generate all those insights on the free tier, but our first tiers of paying users starts at a very affordable, like $60 a month, and you can see all the insights and patterns in your data, and then certain features and certain volumes of predictions push you up into an enterprise tier, which is a little bit more expensive, but still pretty affordable.

**JOHN:** Our plans are all like on a month-to-month basis, and our intent there is to make it really low risk to get started with machine learning. Historically, I think a lot of vendors have asked you to sign up for a year or sometimes multiple years of service before you even are deriving value from the platform.

**JOHN:** We really want it to be the other way around. We want to win collectively with our customer base. So, if we can't get you to value pretty quickly, we don't expect you to keep paying us, and all of that adds up to, I think, the right solution for the longer tail of users. To make it really easy for them to try and play around with and learn and understand the value they could derive from it.

**JOHN:** And then we share a little bit of that win. But often businesses like make massive multiples in return once they get some of these models going. So, our main task for the enterprises we work with is to get them to a model or two that's very valuable to their business. And at that point, our cost is negligible in relation to the gains and wins that they're getting. So that's how the pricing model works. Forces us to really deliver value quickly and makes it like pretty low risk actually to get started with machine learning.

**CRAIG:** There's been a lot of talk about business technologists, people outside of the IT department at companies taking on software tools, I would imagine.

**CRAIG:** That is your target market. People that really understand the business, really understand the data and who are technologically curious and adept enough to use the platform. Is that right? and how do you identify those people in an organization because it's not necessarily the C-suite that's going to be using this stuff.

**CRAIG:** It's people further down in the organization.

**JOHN:** It's the people that the person in marketing or sales goes to or the executive goes to and says, tell me what's going on with blank. And they go work in the data and try and come back with an answer. That's our target customer, their analysts, their business strategy folks who work in data like all the time, but they're not necessarily coders, they're not data scientists.

**JOHN:** we look for businesses that have analytics teams and data strategy teams where they understand the value of the data and have hired people to work against it. Even in businesses with data scientists, it's typically like 10 to one or something. They have 10 analysts working out at the business unit groups helping make data driven decisions for every one data scientist they have working on high scale, product level data problems. We often deliver our end value through them to the business user or the executive who's looking for those insights or patterns from the data itself. And where they roll up varies by business.

**JOHN:** Sometimes those are embedded in IT teams sometimes not. Sometimes they're like matrix out underneath the individual functional areas. Like a marketing team will have an analyst group component underneath it that helps them make their decisions. So, it really is pretty broad depending on the business org structure themselves.

**CRAIG:** I would imagine too that IT departments could use a platform like this because they're getting asked all the time from the business units, to give them something that'll solve a problem. And rather than, having to recreate the wheel, they could use a platform like Akkio create a web application and turn it over to the business units.

**CRAIG:** Do you see that?

**JOHN:** Yeah, we see that sometimes. So, there's a couple benefits to putting machine learning in more people's hands, and one of the key ones is people start to learn how you leverage a tool like machine learning in a business to derive value. And once more people in the org learn that they're able to make better use of it across the org and so there's lots of times where the business unit comes to the IT team or maybe even the data science team, and they ask them to do something that doesn't make a lot of sense because they don't really understand how machine learning works very well. And then there's this whole dialogue of explanation of what's possible or not. And a lot of time is sunk even internally and educating.

**JOHN:** And so, one of the benefits of using a no code ML platform in your business and as spread throughout your business as possible is it helps everyone start to understand the types of questions you can answer with machine learning, how to think about framing the data to ask and answer those questions and how to leverage it.

**JOHN:** And that means you ask more intelligent questions all the way up the chain, and you make more intelligent decisions with your data all the way down into the furthest like extremities of your business. And you add up all of those efficiency enhancements, and it's really game changing for businesses.

**JOHN:** And the ones that adopt it throughout their operations the soonest are going to have major competitive advantages. And then I suspect in a few years most people will be adopting it just to keep up. It definitely is a way to enable a group of people who couldn't self-serve before to learn, to start to self-serve, and then eventually just to actually do the analysis on their own and see the driving factors and make the decisions without running that interrupt loop on your data team or on your IT team?

**CRAIG:** I have one last question. Where does the name Akkio come from?

**JOHN:** Great question. My co-founder Abe who's our CEO, when he started the company, the shortest domain name that he had.

**JOHN:** Was akk.io for acknowledge.in out, and we just leveraged that one because we thought shorter would be better, of course. Then we converted it to Akkio.com and that's where the name is derived from.

**JOHN:** It was just the shortest domain that my co-founder had.

**CRAIG:** That's it for this episode. I want to thank Jon for his time. If you want to learn more about Akkio’s platform, check them out at Akkio.com and try it for free. I also want to thank our sponsor, ClearML, which has a suite of machine learning tools for AI developers. Check them out at clear.ml.

**CRAIG:** As always, you can find a transcript of our conversation today on our website, eye-on.ai.

**CRAIG:** And finally, remember, the Singularity may not be near. But AI is about to change your world, so pay attention.