

Sinan Ozdemir

And why he is cool



About me

- Passionate educator
 - Obtained Masters from and was a lecturer at Johns
 - High school teacher
 - Taught in two different prisons in Maryland
 - Taught in bootcamps like General Assembly over 15 times
- Textbook Author
 - Four textbooks (one of which is Microsoft's book of choice for Data Science)
 - Primary focus on AI, ML, Data Science
 - <https://azure.microsoft.com/en-us/resources/principles-of-data-science/>
- YC Alum with Kylie.ai
- Podcast guest conference speaker
 - Launch, SuperDataScience podcast and conference, Blackhat, and more



Project - Kylie.ai

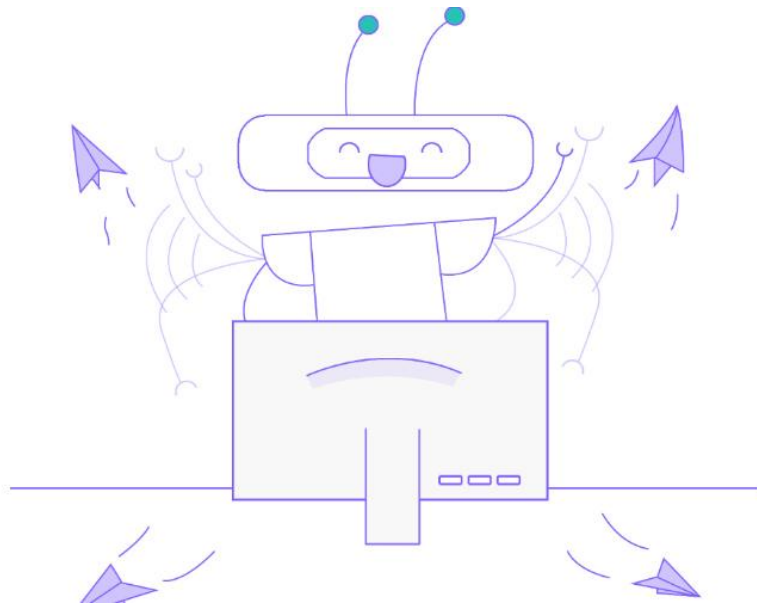
CTO & Founder

ML Engineer, DevOps, Client Manager,
Customer Service, Sales, Compliance
Officer, etc.....

Company Overview

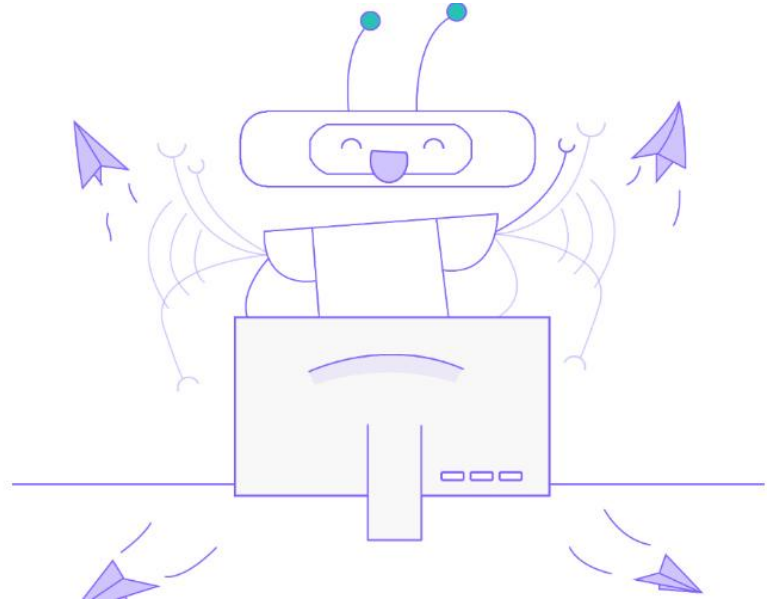
Kylie.ai automates enterprise-grade conversational AI and has serviced customers such as:

- CenturyLink (third largest telco in America)
- Luxottica (owner of SunglassHut, Rayban, Oakley, etc)



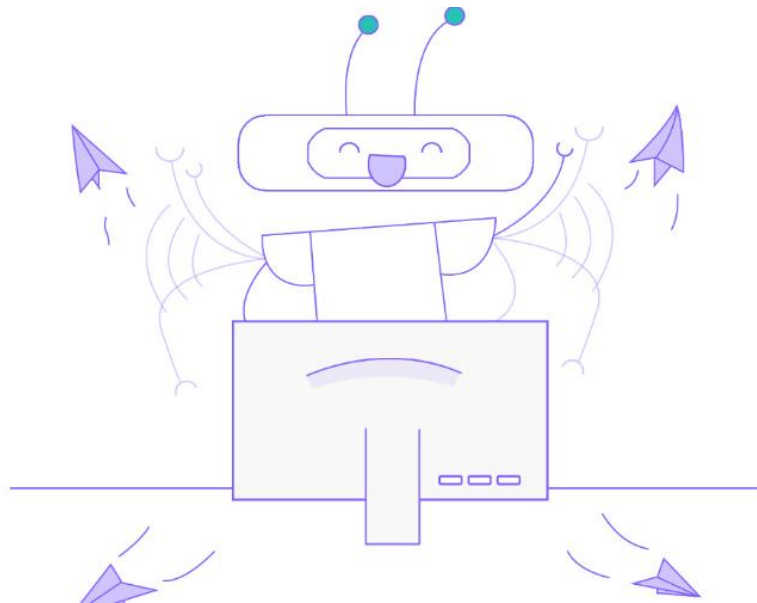
Business Case

- Automate > 25% of conversations
- Omni-channel
 - Voice, SMS (Twilio)
 - Chat (LiveEngage, Intercom, etc)
- Auditability
- Scalability
 - System designed to handle > 250,000 calls a month



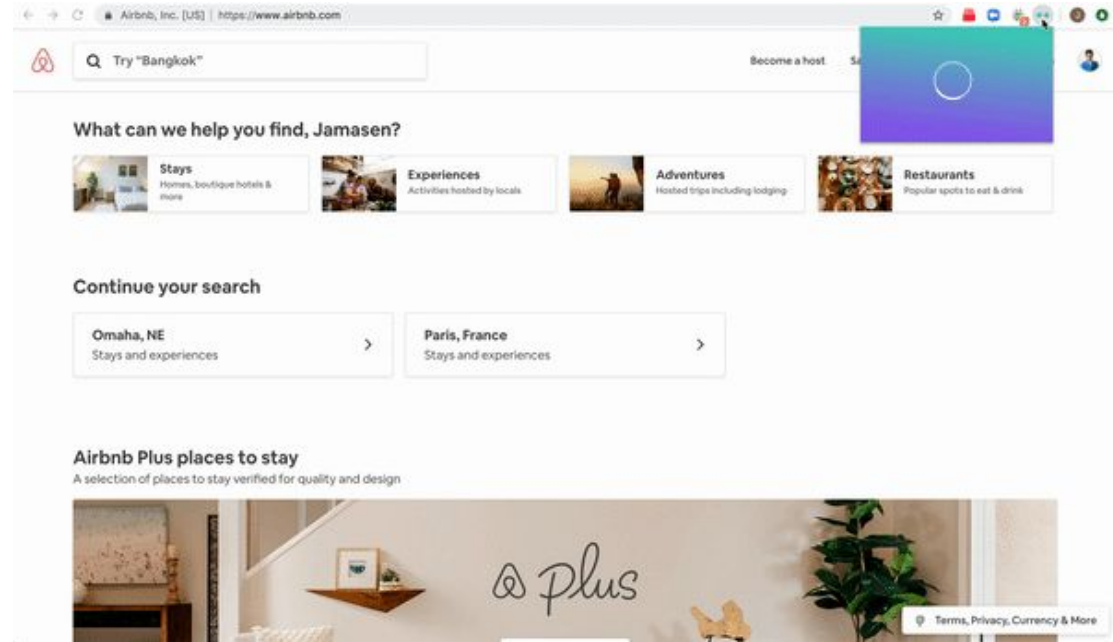
Tech Stack

- DevOps
 - K8s, Docker
 - AWS, EKS, RDS
- ML
 - Scikit-learn
 - GLoVe
 - Tensorflow
- App/API
 - Django
 - React
 - Redis / RabbitMQ



Products

- RPA:
 - Sidekick
- Conversational AI
 - ML Framework
 - SIMon
- Topic Analysis
 - Unsupervised learning of conversation automation



Take Home

https://github.com/sinanuozdemir/airbnb_th/blob/master/Take.Home.ipynb

General Approach

- Treat it like a classification problem
 - Features were listing criteria, query criteria, preference criteria (dynamically generated)
 - Response was label column
 - host_contact, book, click, impression or a combined contact and book
- Steps
 - EDA / Preprocessing
 - Feature Engineering
 - Machine Learning Pipeline building
 - Training
 - Evaluation

Attempts

- Null model (random model): predicts label at the same distribution as the ground truth
 - Custom precision: ~ 21-25% (not consistent because..it's random)
- First approach: use listing, query to predict label
 - Sometimes barely better than random model: 23%
- Add in preference vector (a vector with the same number of cols as listing that represents the user's preference
 - Calculated by taking previous clicks, contacts, books before `ts_search` and multiplying them together
 - Custom precision: ~ 27%
 - Biggest problem is explode number of cols
- Used the dot product between preference and listing to create a single "preference_listing" col and got rid of preference col as well as query cols
 - Best precision yet: ~ 31%