AI = Your Data

(And what that means for you as you build a virtual assistant)

Why do you bother working on chatbots?

Why do you bother working on chatbots?

Subtext: I think most conversational AI projects are bad

What makes a bad Conversational AI project?

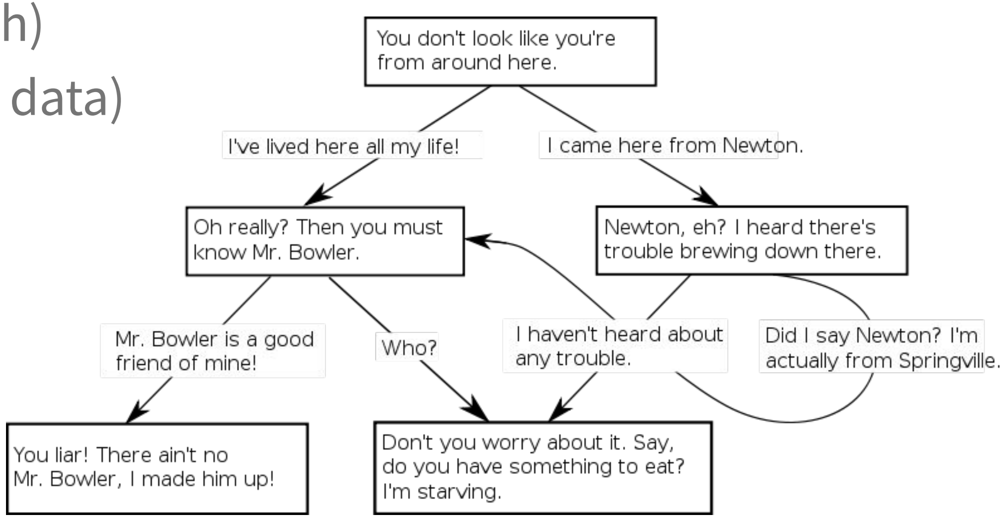
It doesn't help people do what they need to do.

How do you know what people need to do?

- You make an educated guess
- You ask them (UX research)
- They tell you (You look at data)

How do you know what people need to do?

- You make an educated guess
    - Complete top-down design (state machines, dialog trees) can be a good approach, but they're inflexible
- You ask them (UX research)
- They tell you (You look at data)



You don't look like you're from around here.

I've lived here all my life!

I came here from Newton.

Oh really? Then you must know Mr. Bowler.

Newton, eh? I heard there's trouble brewing down there.

Mr. Bowler is a good friend of mine!

Who?

I haven't heard about any trouble.

Did I say Newton? I'm actually from Springville.

You liar! There ain't no Mr. Bowler, I made him up!

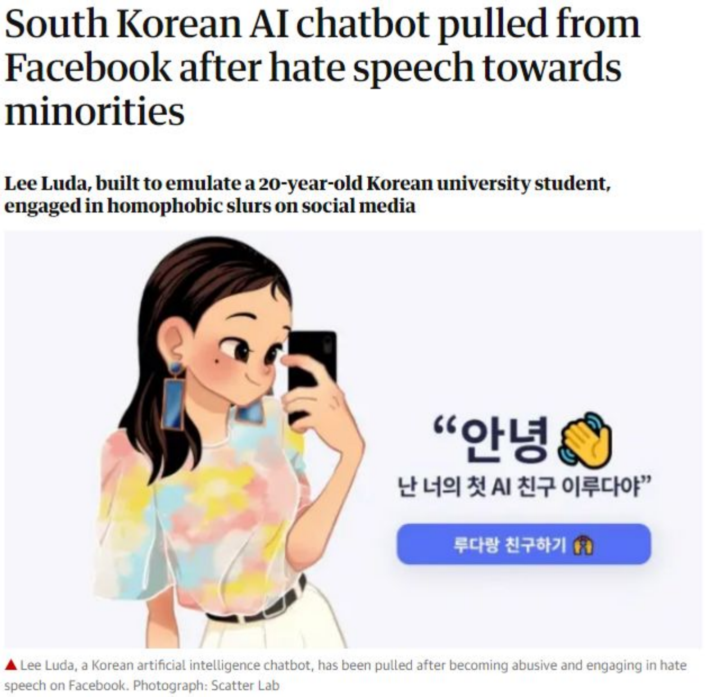
Don't you worry about it. Say, do you have something to eat? I'm starving.

How do you know what people need to do?

- You make an educated guess
- ✨You ask them (UX research)✨
    - Not covered today but 100% do it if you can
- They tell you (You look at data)

How do you know what people need to do?

- You make an educated guess
- You ask them (UX research)
- They tell you (You look at data)
  - More flexible
  - But don't assume that all you need is more user-generated data!



South Korean AI chatbot pulled from Facebook after hate speech towards minorities

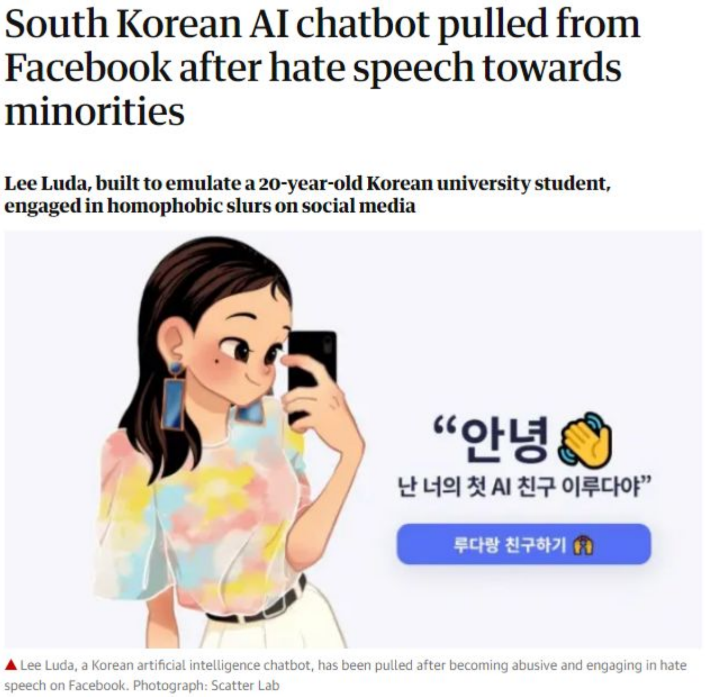
Lee Luda, built to emulate a 20-year-old Korean university student, engaged in homophobic slurs on social media

▲ Lee Luda, a Korean artificial intelligence chatbot, has been pulled after becoming abusive and engaging in hate speech on Facebook. Photograph: Scatter Lab

https://www.theguardian.com/world/2021/jan/14/time-to-properly-socialise-hate-speech-ai-chatbot-pulled-from-facebook

How do you know what people need to do?

- You make an educated guess
- You ask them (UX research)
- They tell you (You look at data)
    - More flexible
    - But don't assume that all you need is more user-generated data!
    - Happy medium:
        - your system learns from data
        - you provide additional structure and organization



South Korean AI chatbot pulled from Facebook after hate speech towards minorities

Lee Luda, built to emulate a 20-year-old Korean university student, engaged in homophobic slurs on social media

▲ Lee Luda, a Korean artificial intelligence chatbot, has been pulled after becoming abusive and engaging in hate speech on Facebook. Photograph: Scatter Lab

https://www.theguardian.com/world/2021/jan/14/time-to-properly-socialise-hate-speech-ai-chatbot-pulled-from-facebook

What is “data” in conversational AI?

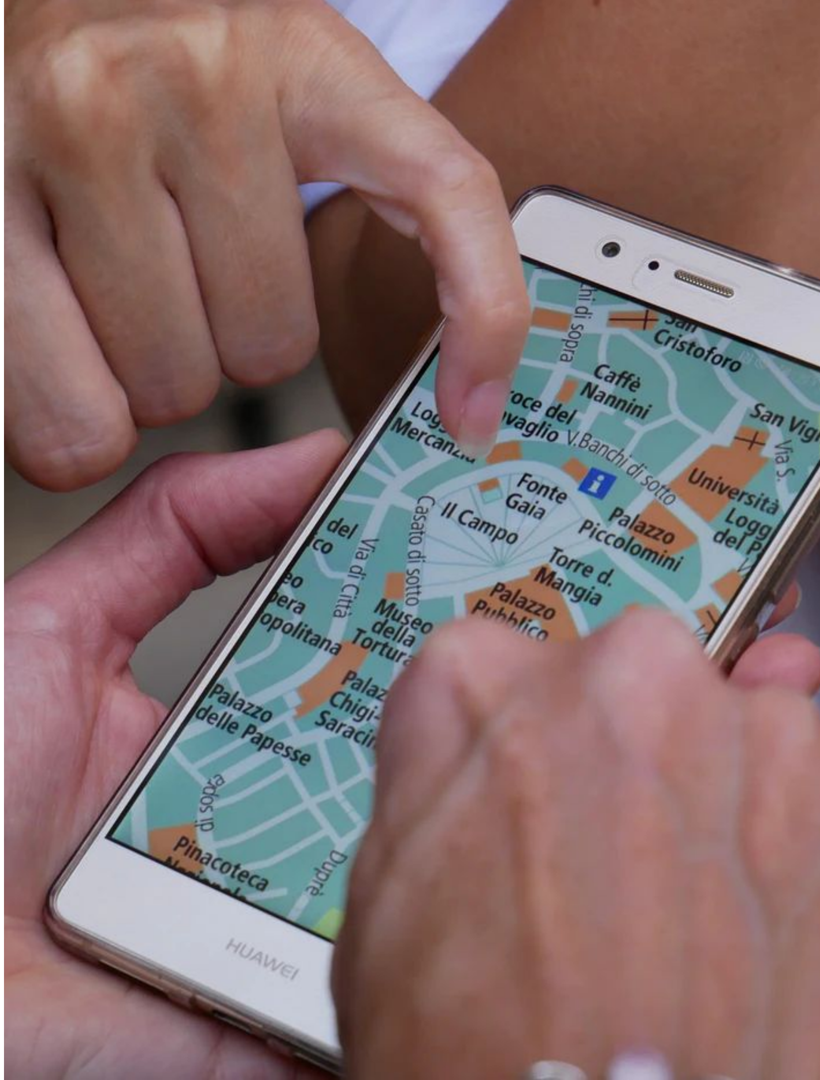
- The text data use to pretrain any models or features you're using (e.g. language models, word embeddings, etc.)
- User-generated text
- Patterns of conversations
- Examples:
  - Customer support logs (assuming data collection & reuse is covered in your privacy policy)

Two different assistants can have more or less the same underlying ML code. (That’s what makes building a conversational AI framework possible!)

What makes your assistant work for you and your users is **your data & how you structure it.**

- **Curate** = decide what data to use as you train & retrain your assistant
- **Annotate** = apply (or correct) labels for individual pieces of data

- Intents
  - If you already have data
  - If you don't
- Stories
- Checking if it works



Intent = something a user wants to do

Intent = something a user wants to do

Quick test: is this a VERB (inform, book_trip, confirm)?

If you have data

- Modified content analysis:
  - Go through data (or sample) by hand and assign each datapoint to a group
  - If no existing group fits, add a new one
  - At given intervals, go through your groups and combine or seperate them as needed
  - Start with 2-3 passes through your dataset
- Can't you just automate this?
  - Maaaybe, but I wouldn't recommend it: no guarantee clusters will map well to user needs

(Even) If you don't have data

- Start with the most common intent
  - Most people want to do the same thing
  - Use the experts in your institution (e.g. support staff)
- Start with the smallest possible number of intents (that cover your core use case)
- Everything else goes in an out of scope intent
  - If your assistant can't handle something, give users an escape hatch right away
- Additional intents will come from user data

Why fewer intents?

- Older style of conversational design:
    - You need an intent for everything your user might want to do!
- Rasa style CDD:
    - You only need to start with the most popular, important intents & a way to handle things outside them
    - Continue to build from there if that’s what users need

Why fewer intents?

- Human reasons
  - More intents = more training data, maintenance, documentation
  - More intents = annotation more difficult
- ML reasons
  - Transformer classifiers scale linearly with the # of classes*
  - Entity extraction (esp. with very lightweight rule-based systems like Duckling) is often faster

*Taming pretrained transformers for eXtreme multi-label text classification, Chang et al 2020 @ KDD

Paring down intents

- Don’t use intents as a way to store information
    - Storing information = slots
- Do a lot of the same tokens show up in training data for two intents? Consider if they can be combined
- I would personally start with:
    - max 10 intents
    - min 20 training examples (for each intent)

book_train:

- One train ticket
- Need to book a train ride
- A rail journey please

book_plane:

- One plane ticket
- Need to book a plane ride

make_booking:

- One train ticket
- Need to book a train ride
- A [rail](train) journey please
- One [plane](air) ticket
- Need to book a [plane](air) ride
- i'd like to book a trip
- Need a vacation

Training data for an intent

- User-generated > synthetic
- Chat-based interactions tend to be informal
- Each utterance should unambiguously match to a single intent
    - You can verify this using human sorting & inter-rater reliability
- Is an utterance ambiguous?
    - Use end-to-end instead (the raw text as training data w/out classifying it)

Unambiguous:

- Hi there
- Hieeeeeeeeeeee
- Hola
- I said, helllllloooooO!!!!
- What is up?
- ayyyy whaddup
- hello robot
- hello sara
- merhaba
- ola sara

Ambiguous (goes in end to end):

- good day
- ciao
- alhoa 🤙

Stories = training data to decide what your assistant should do next

Stories = training data to decide what your assistant should do next

Older style: Most of the design work

Newer style: Provide examples & system extrapolates from them

Stories

- If you have conversational data:
  - If you have conversations, start with the patterns you see in them
  - Find a new intent? Add it to your intents
- Generating your own conversational patterns:
  - It's easiest to use interactive learning to create stories (in command line or Rasa X)
  - Start with common flows, “happy paths”
  - Then add common errors/digressions
- Once your model is trained:
  - Add user data for more ASAP

Stories

- If you have conversational data:
  - If you have conversations, start with the patterns you see in them
  - Find a new intent? Add it to your intents
- Generating your own conversational patterns:
  - It's easiest to use interactive learning to create
    command line or Rasa X)
  - Start with common flows, “happy
  - Then add common errors/digressio
- Once your model is trained:
  - Add user data for more ASAP

Do you need **conditional logic** in your conversations? Like: "If they're already signed in don't ask for a name" or "never show account balance without a PIN?

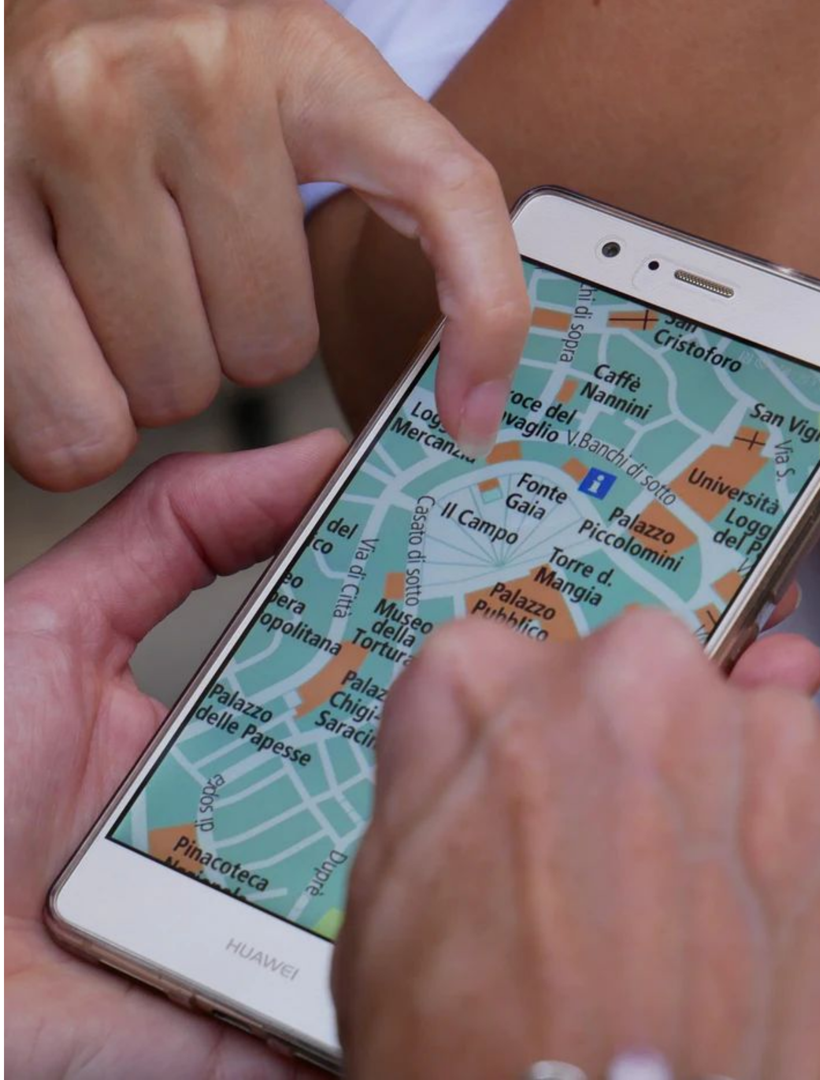
Add rules!

How do you know if it works?

- Reviewing user conversations!
- Tests
  - Sample conversions that should always be handled the same
  - Good to shoot for 100% correct
- Validation
  - Checking that your model can guess correctly at an acceptable rate
  - Be very suspicious of accuracy near 100% 🤔

Takeaways

- Language data is what makes your Rasa assistant work
- Providing structure for language data is the first step for building NLP systems
- Start with the fewest possible, most popular things
- Get your prototype in front of users ASAP



Thanks! Questions?

@rctatman