



Booking.com

The NLU Orchestra

Amit Beka | 12.12.2019

Photo by [Karim MANJRA](#) on [Unsplash](#)



Booking.com.



**Biggest accommodation
website worldwide**

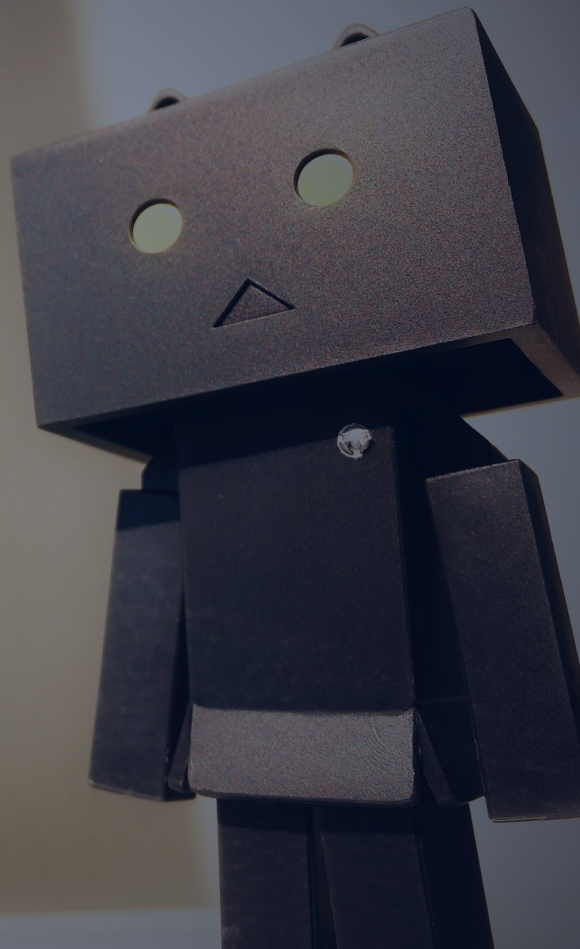


**Chatbot on Facebook
Messenger**



**Try us:
@bookingcom**

Booking.com



Myself.

Senior ML Engineer @ Booking.com

15+ years of coding

7 years of crafting chatbots

About this talk.

- The following is based on actual events, bugs and challenges
- Not only chatbots
- Build any complex NLU product



Linear Pipeline.



Hotel in Warsaw
CITY

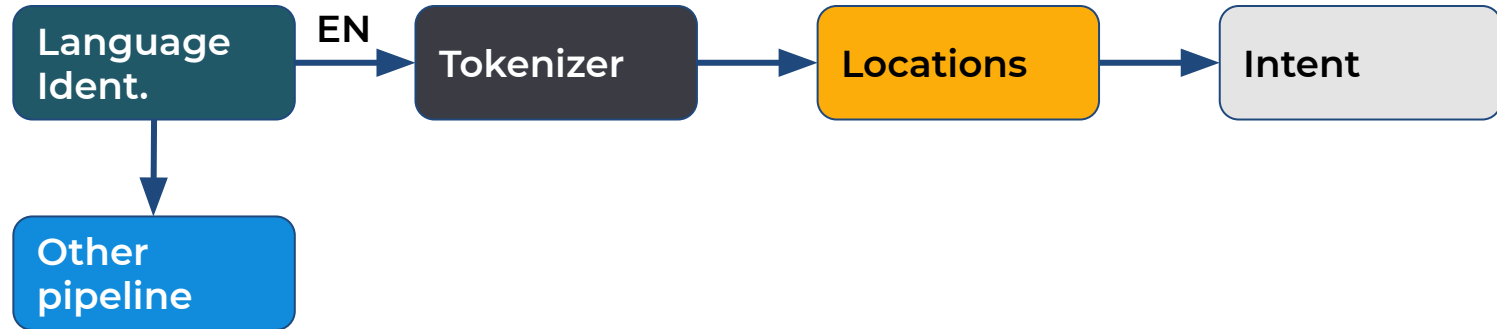
Hotel
Search



Almost Linear Pipeline.



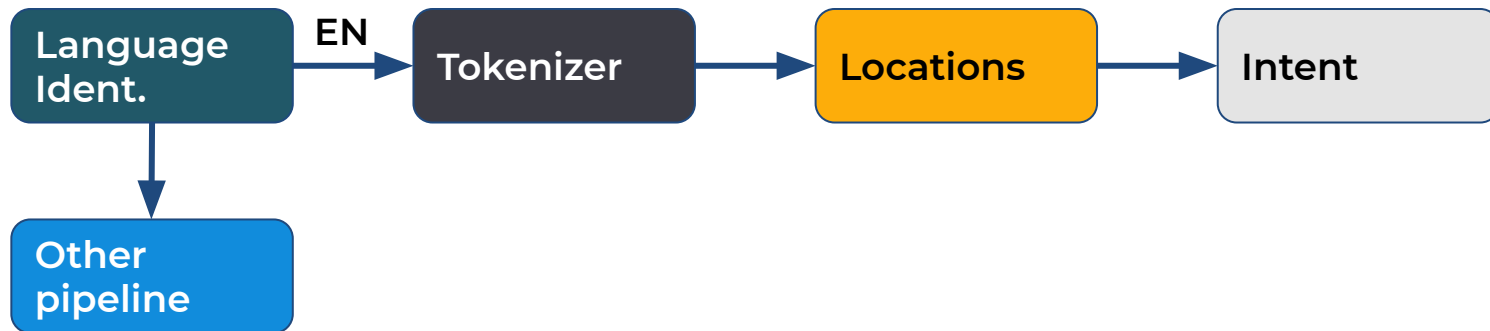
Albergo a Varsavia





Cancel eden hotel
CITY

Cancel
Room

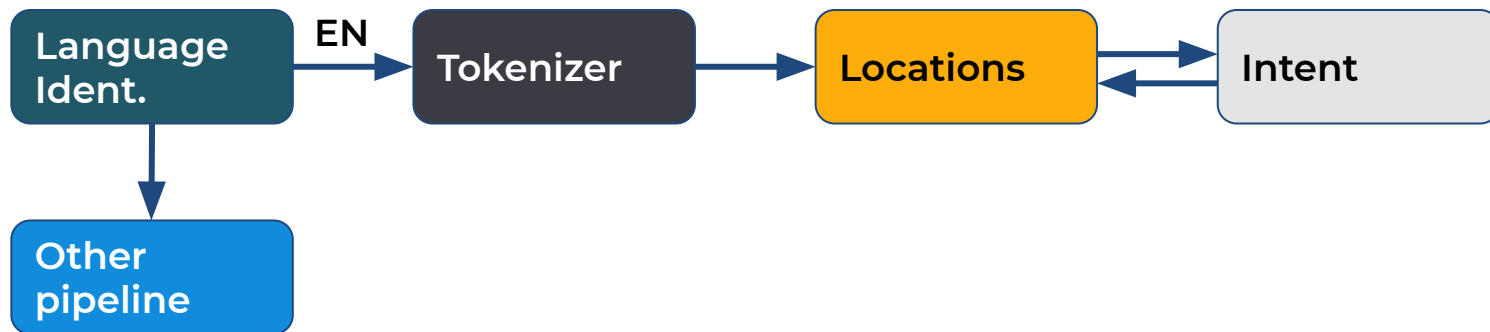


Feedback loop.



Cancel eden hotel
HOTEL

Cancel
Room

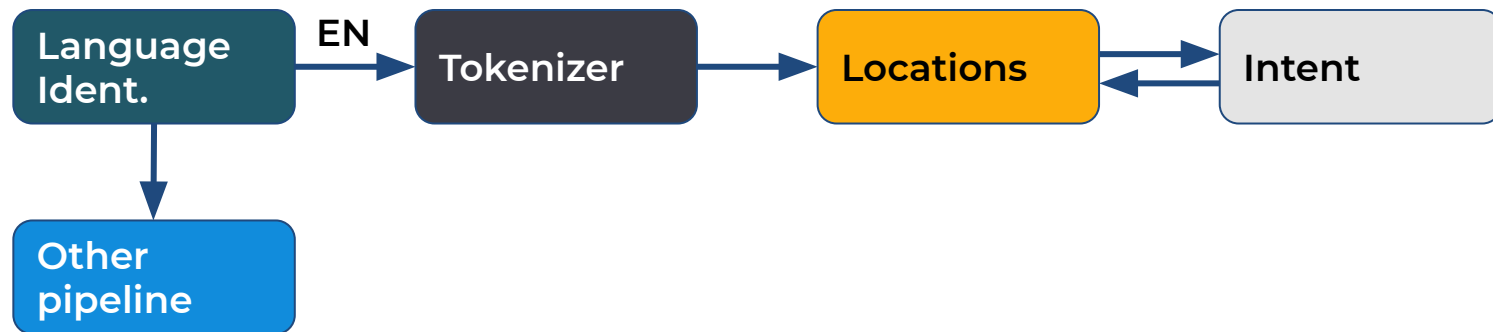




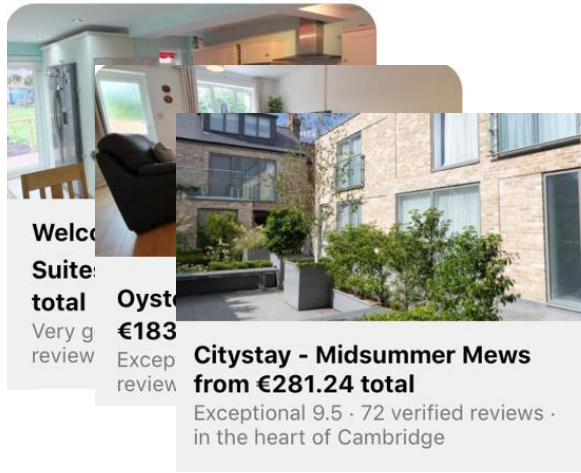
Hotel in Cambridge

CITY

Hotel
Search

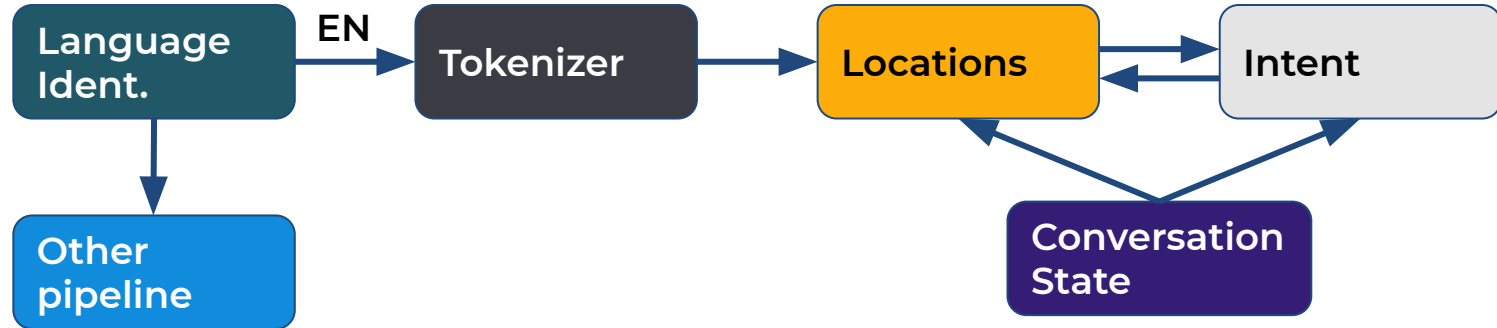


Conversation ≠ Pipeline.



Massachusetts
STATE

Fix
Location

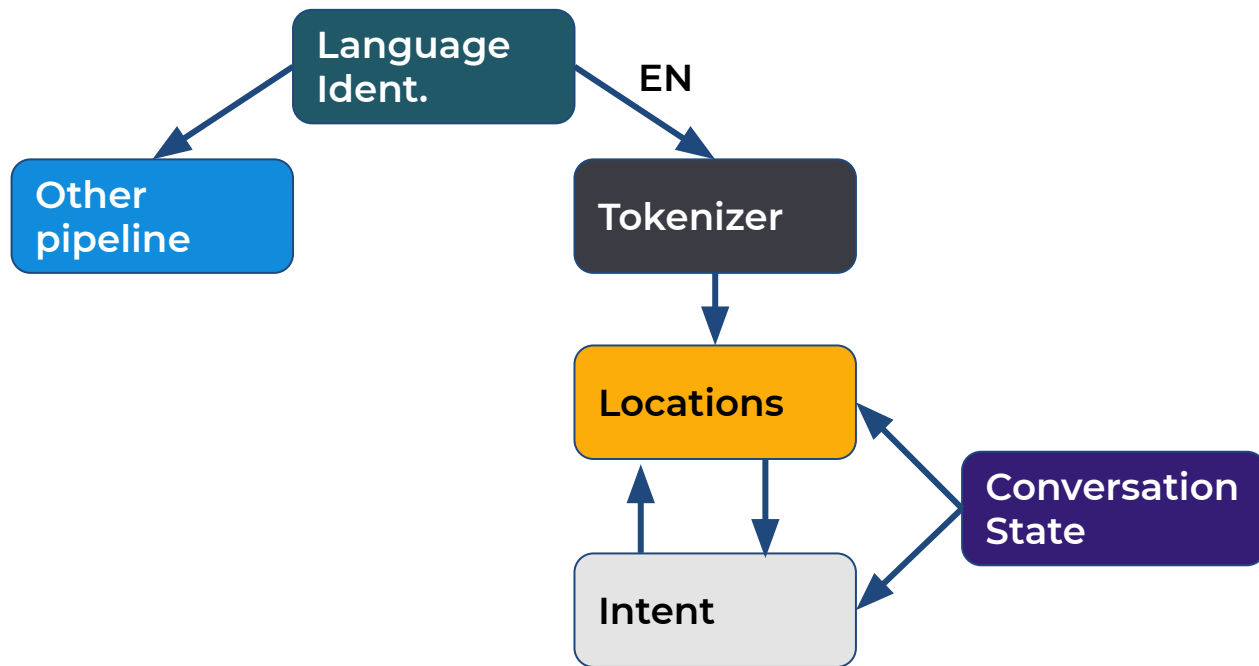




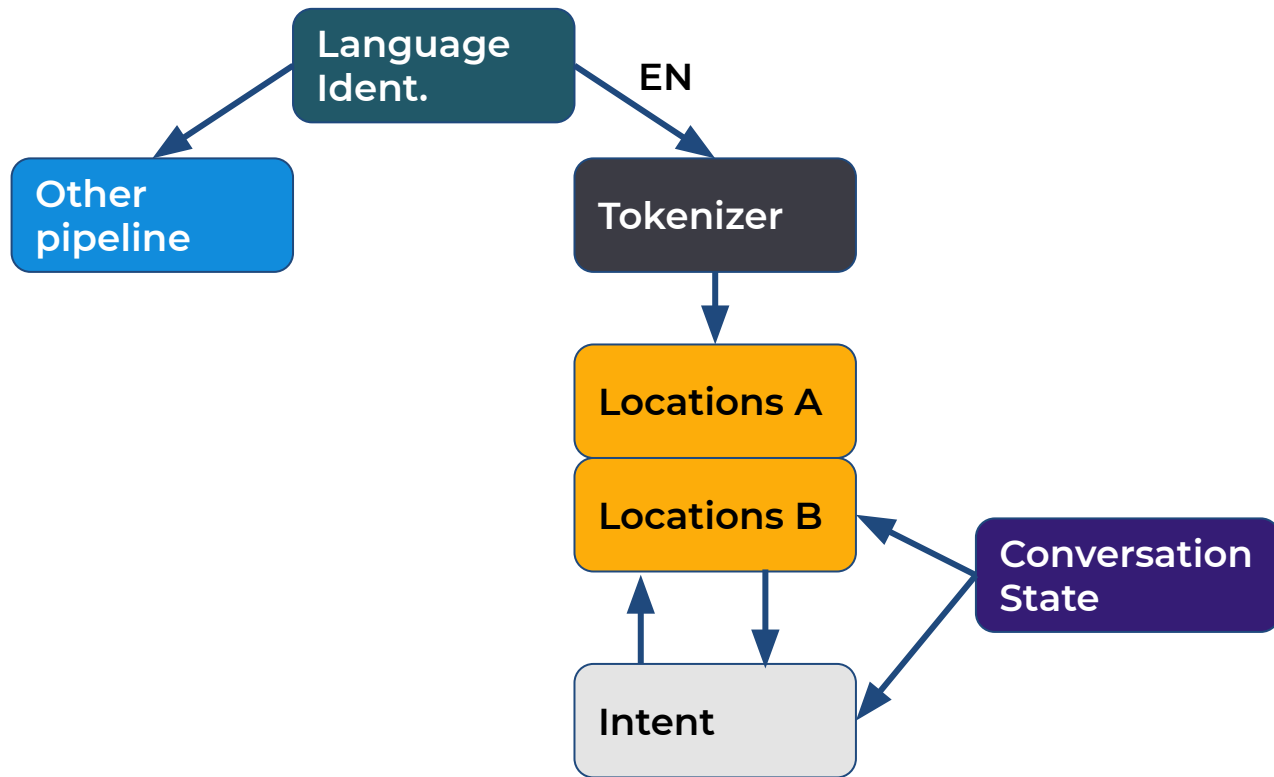
Problems.

- **Conversations are non-linear**
- **Pipelines are inherently broken**
- **Multiple dependencies per model**
- **Same “component” called many times, with different inputs**

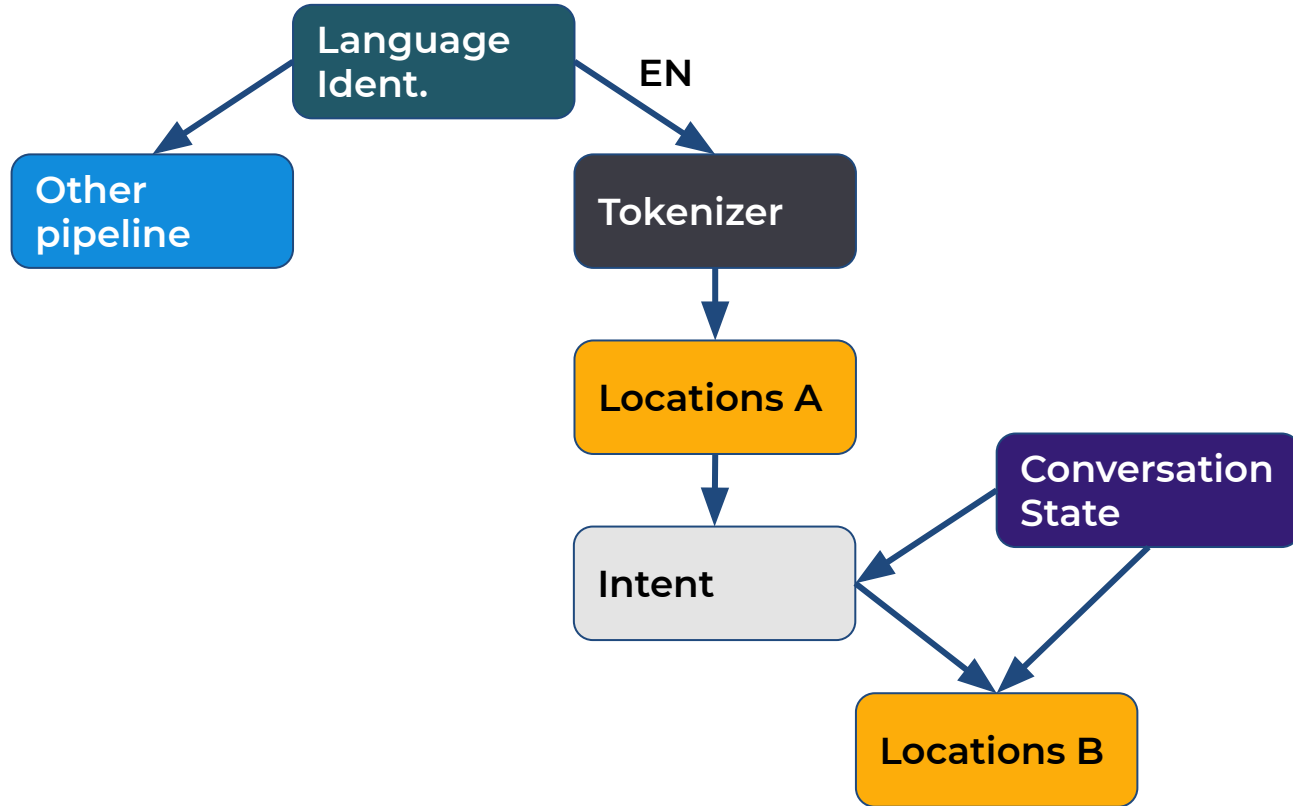
Maybe as a graph?



Split the job.



Directed Acyclic Graph!



Proposed Solution



Separate

Tasks are broken to separate models if they have different dependencies

Locations from tokens

Locations from intent

Locations from conversation

Proposed Solution



Define

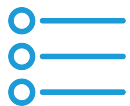
A component is defined by its dependencies, conditions and output type

Depends on: english_tokenizer,
locations_simple,
conversation_state
Only if: not_first_message

Intent to fix previous

Outputs: intent (label)

Proposed Solution



API

Standard output/input for each component: tokens, spans, labels and state

Tokens: words with possible attributes

Spans: consecutive tokens with type and value

Labels: key-value pairs applying to the whole text

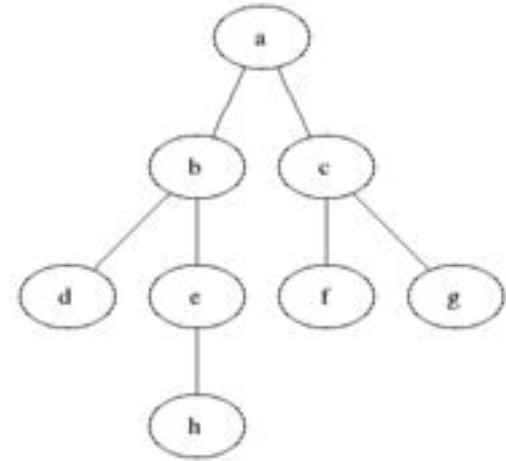
State: previous messages, metadata

Proposed Solution



Orchestrate

A conductor runs each model in its turn according to the dependency graph



Advantages



**Microservice
architecture**



**Unit test each
component**

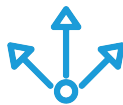


**Track errors to their
specific component**

Advantages



Reuse good models



**Flexibility to extend
and use different
sub-graphs**



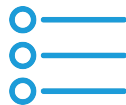
**Experimentation
friendly**

Summary



Separate

Tasks are broken to separate models if they have different dependencies



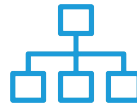
API

Standard output/input for each component: tokens, spans, labels and state



Define

A component is defined by its dependencies, conditions and output type



Orchestrate

A conductor runs each model in its turn according to the dependency graph

Thank you!

www.linkedin.com/in/amitbeka

Booking.com

Photo by [Gabriel Santos Fotografia](#) from [Pexels](#)