Specialist Diploma in Artificial Intelligence

AI Applications with Deep Learning

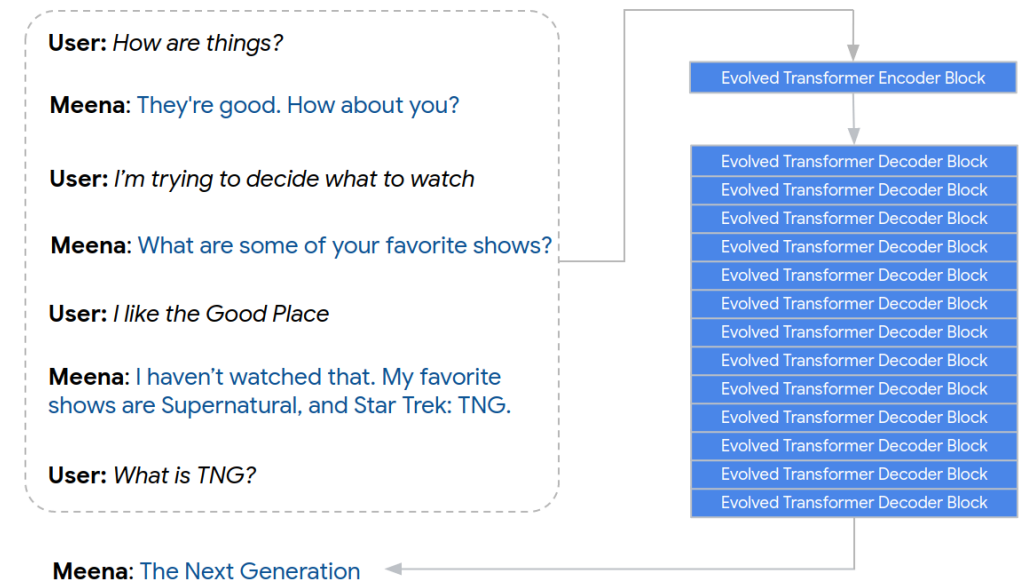# Creating Your Own Chatbot

Sub topics

- **What is a Chatbot?**
- **Transformers with the HuggingFace Library**
- **Intent Recognition**
- **Custom Named Entity Recognition**
- **Putting It Altogether**
- **Other Advances in Natural Language Processing**

Creating Your Own Chatbot

WHAT IS A CHATBOT?

Open Domain Chatbot

- ## Multi-Turned Open Domain Chatbots
  - Meena Chatbot by Google
  - Uses seq2seq architecture
  - Has 2.6B parameters.
  - Trained on 341GB dataset of social media conversations

User: *How are things?*

Meena: They're good. How about you?

User: *I'm trying to decide what to watch*

Meena: What are some of your favorite shows?

User: *I like the Good Place*

Meena: I haven't watched that. My favorite shows are Supernatural, and Star Trek: TNG.

User: *What is TNG?*

Meena: The Next Generation

Evolved Transformer Encoder Block

Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block
Evolved Transformer Decoder Block

https://ai.googleblog.com/2020/01/towards-conversational-agent-that-can.html

Google's Meena Chatbot

https://www.youtube.com/watch?v=cA0vHYzGR8s

A Simple Chatbot – Closed Domain

- What we will try to build:
  - Closed Domain = Chatbot meets specific business objectives

- For each text said by the user:
  - Use AI to Classify Intent (Intent Classification)
  - Use AI to Extract Entities (Entity Recognition)
  - Write Rules to Process Intent and Entities
  - Write Code to Display Reply to User

Creating Your Own Chatbot

TRANSFORMERS WITH HUGGINGFACE LIBRARY

Recall...

- The HuggingFace library:
  - A popular Deep Learning Natural Language Processing library
  - Includes many variations of the Transformer architecture
  - Includes many pipelines for various Natural Language Processing tasks
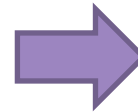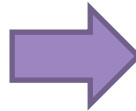 - Can run on-premise (non-cloud)

Supported NLP Tasks

- ## Sequence Classification
  Topic modelling, text classification, positive/negative sentiment analysis, intent recognition
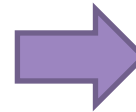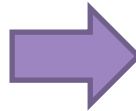
"This is a great movie" ➡ 😊 ➡ **POSITIVE**

Supported NLP Tasks

- Extractive Question & Answer

  Answers a question by extracting an answer from FAQs, customer reviews, social media posts, etc.

"How do I contact the sales team?" ➡️ ➡️ "**You can call 011-2233-4444**"
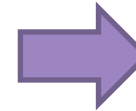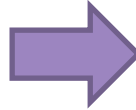
The blue text is generated as a reply by the transformer. This is likely extracted from a paragraph of text in its training data
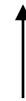
Supported NLP Tasks

- ## Text Generation
  Generates a human-like text that keeps the writing as coherent as possible

"As far as I am concerned, I will" → → "As far as I am concerned, I will **be the first to admit that I am a not fan of the idea of…**"

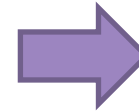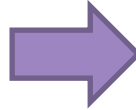The blue text is generated by the transformer

Supported NLP Tasks

- ## Named Entity Recognition
  Classify each word or token in a sentence to identify what parts of speech, or what properties they belong to.

"Hugging Face Inc. is a company based in New York City"

→

"Hugging Face Inc."
"New York City."

ORGANIZATION
LOCATION

The transformer decides which parts of the text belong to certain entities

Supported NLP Tasks

- ## Summarization

  Summarizes a paragraph of text.

New York (CNN) When Liana Barrientos was 23 years old, she got married in Westchester County, New York. ... A year later, she got married again in Westchester County, but to a different man and without divorcing her first husband. ... Only 18 days after that marriage, she got hitched yet again. Then, Barrientos declared "I do" five more times, sometimes only within two weeks of each other. ... In 2010, she married once more, this time in the Bronx. In an application for a marriage license, she stated it was her "first and only" marriage. ... Barrientos, now 39, is facing two criminal counts of "offering a false instrument for filing in the first degree," referring to her false statements on the ... 2010 marriage license application, according to court documents. ... Prosecutors said the marriages were part of an immigration scam. ... On Friday, she pleaded not guilty at State Supreme Court in the Bronx, according to her attorney, Christopher Wright, who declined to comment further. ... After leaving court, Barrientos was arrested and charged with theft of service and criminal trespass for allegedly sneaking into the New York subway through an emergency exit, said Detective ... Annette Markowski, a police spokeswoman. In total, Barrientos has been married 10 times, with nine of her marriages occurring between 1999 and 2002. ... All occurred either in Westchester County, Long Island, New Jersey or the Bronx. She is believed to still be married to four men, and at one time, she was married to eight men at once, prosecutors say. ... Prosecutors said the immigration scam involved some of her husbands, who filed for permanent residence status shortly after the marriages. ... Any divorces happened only after such filings were approved. It was unclear whether any of the men will be prosecuted. ... The case was referred to the Bronx District Attorney\'s Office by Immigration and Customs Enforcement and the Department of Homeland Security\'s ... Investigation Division. Seven of the men are from so-called "red-flagged" countries, including Egypt, Turkey, Georgia, Pakistan and Mali. ... Her eighth husband, Rashid Rajput, was deported in 2006 to his native Pakistan after an investigation by the Joint Terrorism Task Force. ... If convicted, Barrientos faces up to four years in prison. Her next court appearance is scheduled for May 18.
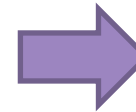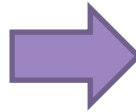
Liana Barrientos, 39, is charged with two counts of "offering a false instrument for filing in the first degree" In total, she has been married 10 times, with nine of her marriages occurring between 1999 and 2002. She is believed to still be married to four men.

Supported NLP Tasks

- Translation

Takes a text in one language and translates to another language.

"Hugging Face is a technology company based in New York and Paris"
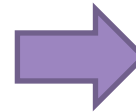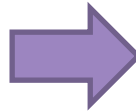
"Hugging Face ist ein Technologieunternehmen mit Sitz in New York und Paris."

Using a 3rd-Party Library

- Using third party framework / tools are common when applying Deep Learning to your problem statement

- Importantly, we need to know how to use that framework to:
  - Construct our training / test data to the model for training
  - Construct an input to the model for prediction / inferencing

Creating Your Own Chatbot

INTENT RECOGNITION – ZERO SHOT CLASSIFICATION

Text Classification

- Intent recognition can be achieved with "Sequence / Text Classification".

- What we are actually doing:

"Some text that you pass in" → **Text Classification** → **WHAT IS IT?**
A classification of what the input text is
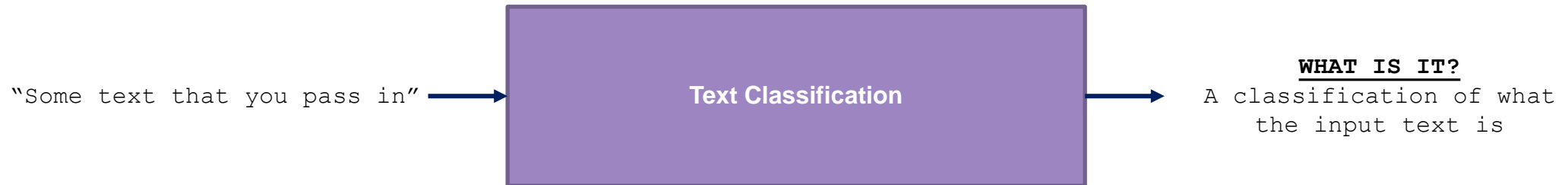
Some Use Cases

Social Media Monitoring
(Complaints, Threats)
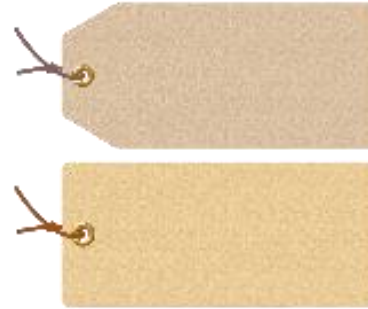
**Automated Tagging of
User Content**
(Tag Suggestions, Categorization)

**Support Ticket Classification /
Intent Recognition**

**Document Classification
and Search**

Sentiment Analysis
(Branding, Public Relations)

Dataset

- We want to build a chatbot that interacts with customers on **flight enquiries** and other flight-related matters.

- Assuming that we use a combination of the following datasets:

  - https://www.kaggle.com/hassanamin/atis-airlinetravelinformationsystem
    ATIS that captured customers' requests and classified each of them

  - https://github.com/alyssaong1/botframework-smalltalk/blob/master/smalltalkkb.tsv
    A small talk chatbot dataset with greetings, yes/no answers and other small talk

Two Ways to Do This

- ## Zero-Shot Text Classification
  Zero-shot means:
  Use a pre-trained model with no *additional* training data for any generic task
  (this task was never seen in the original training data set).

  This is good if you have no training data. Less accurate than training with your own custom dataset
  https://twitter.com/huggingface/status/1293240692924452864?lang=en

- ## Collect Data and Perform Transfer Learning on the Pre-Trained Model with your Custom Labels

Transformers with HuggingFace – Zero-Shot

"Can I know what flights are
available?"

"flight"-related?

"Can I know what flights are
available?"

"greeting"-related?

Zero-Shot Classification Pipeline
(Hugging Face)

Zero-Shot Classification Pipeline
(Hugging Face)

0.9

0.1
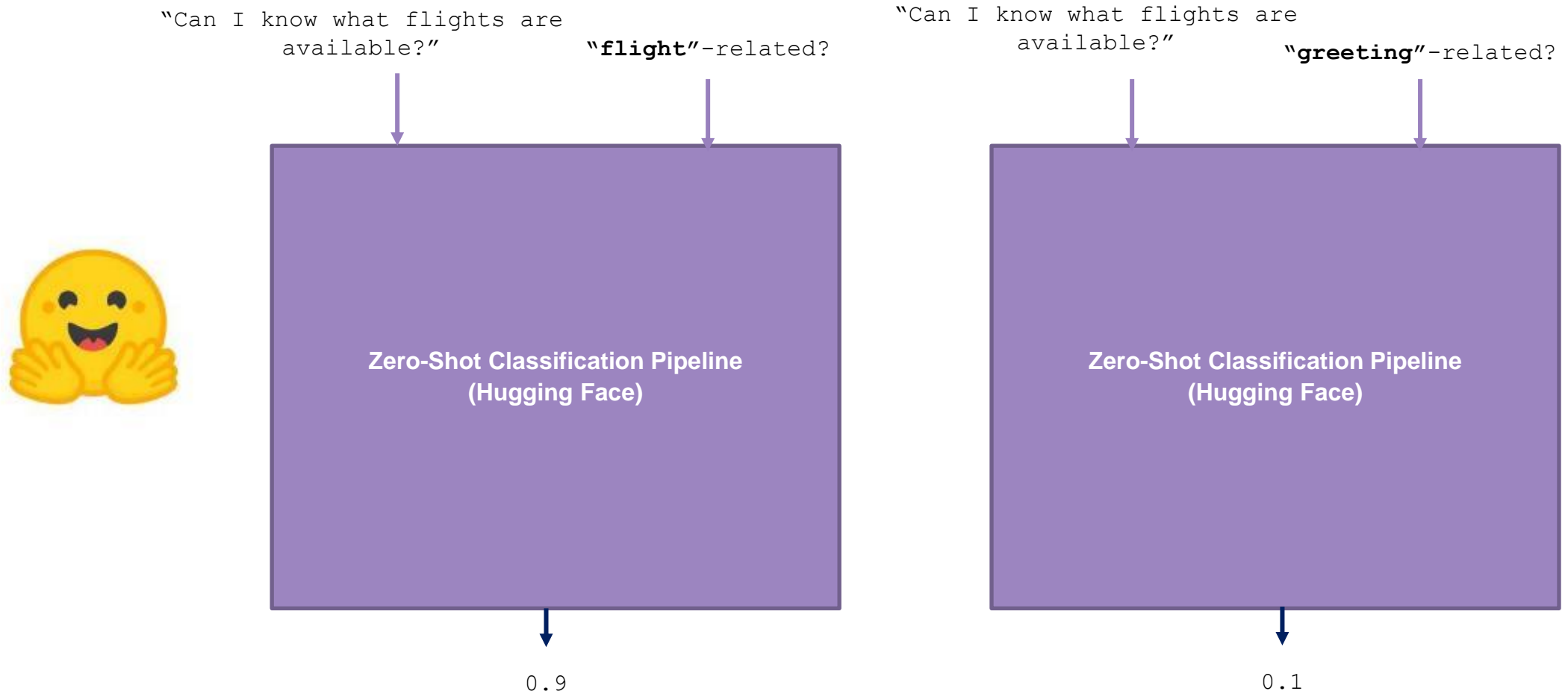
Highest score wins

Transformers with HuggingFace – Zero-Shot

Let's Try!!

https://huggingface.co/zero-shot/

Go ahead and try different sets of broad, logically separate topics:

1. science, mathematics, English
2. politics, sports, finance
3. artificial intelligence, biology
4. happy, sad, neutral, angry

With Zero-Shot Learning

- The candidate "label" are English labels, and the model tries to compute the similarity the input text's meaning with the label's meaning.

- So to achieve good results, the choice of the candidate labels (with clear semantic boundary with the rest of the labels) is very important.

24

Creating Your Own Chatbot

INTENT RECOGNITION – TRAINING WITH CUSTOM LABELS

Transformers with HuggingFace

Custom Trained Labels

"Can I know what flights are available?"

↓

**Transformer Model
(Hugging Face)**

This is what we want to achieve:
- Pass in a sentence as input,
- Retrieve a classification as output

```
chat_abbreviation  : 0.0
chat_flight        : 0.8
chat_yes           : 0.1
others             : 0.1
```

Transformers with HuggingFace

Custom Trained Labels

"Can I know what flights are available?"

↓

DistilBERT Tokenizer

↓	↓	↓	↓	↓	↓	↓	↓	↓
[CLS]	can	i	know	what	flights	are	available	[SEP]

This happens when you call **tokenizer(input_text)**

DistilBERT Tokenizer (Encoder)

101	2064	1045	2113	2054	7599	2024	2800	102

Input word dictionary IDs

1	1	1	1	1	1	1	1	1

Attention Mask

DistilBERT Model

SentenceClassifier

Fully-connected neural network with Softmax activation for classifying **each token** individually

```
chat_abbreviation : 0.0
chat_flight       : 0.8
chat_yes          : 0.1
others            : 0.1
```

In theory, you can classify any sentence to any category, provided you have sufficient data to train your transformer model

Transformers with HuggingFace

Custom Trained Labels

"Can I know what flights are available?"

DistilBERT Tokenizer (Encoder)

This happens when you call **tokenizer(input_text)**

| 101 | 2064 | 1045 | 2113 | 2054 | 7599 | 2024 | 2800 | 102 | Input word dictionary IDs |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Attention Mask |

DistilBERT Model

SentenceClassifier

Fully-connected neural network with Softmax activation for classifying **each token** individually

```
chat_abbreviation : 0.0
chat_flight       : 0.8
chat_yes          : 0.1
others            : 0.1
```

Training and Test Dataset

- We are going to use all of the atis_intent.csv data, but only part of the smalltalkkb.tsv dataset.

- We will construct the dataset manually with Excel to hold the following:
  - All of atis_intent.csv
  - Only greetings / yes / no from smalltalkkb.tsv

Training and Test Data Format

- The format required for the training and evaluation:

- X (inputs):
  - Output from the tokenizer (input word IDs, and attention mask)

- Y (output):
  - Index to the label
    (training: *not* one-hot encoding, inference: one-hot encoding)

Inference

Original text `can i get a flight to new york tomorrow?`

DistilBERT Tokenizer (Encoder)

Data to be fed into model for inference:

Word dictionary IDs `[ 101, 2064, 1045, 2113, 2054, 7599, 2024, 2800, 102]`

Attention Masks `[   1,    1,    1,    1,    1,    1,    1,    1,    1]`

DistilBERT Model + Sequence Classifier

Output logits before softmax (1 value per label, highest is max) `[-2.46, 8.31, -1.35, -1.60,...]`

argmax

`1`
`Label: chat_flight`
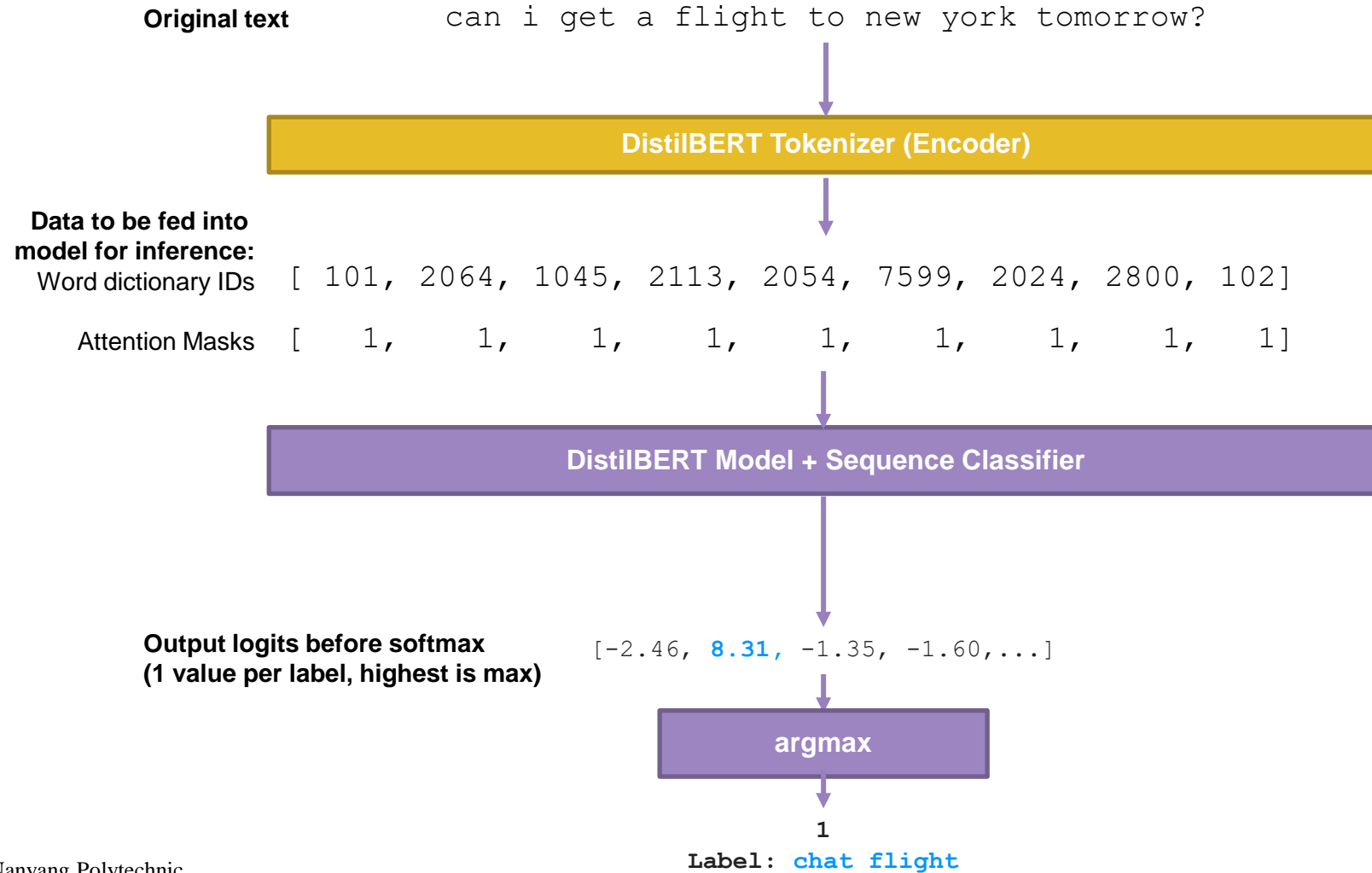
Creating Your Own Chatbot

CUSTOM NAMED ENTITY RECOGNITION

What is Named Entity Recognition?

Seeks to locate and classify named entity mentions in unstructured text into pre-defined categories such as:

the person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc.

https://en.wikipedia.org/wiki/Named-entity_recognition

What is Named Entity Recognition?

Jim bought 300 shares of Acme Corp. in 2006.

Person   Number   ----------- Company -----------   Year

DEMO

https://cloud.google.com/natural-language

What is Named Entity Recognition?

Jim bought 300 shares of Acme Corp. in 2006.

Named Entity Recognition Classifier

Person   ???   Number   ???   ??? ----------- Company -----------   ???   Year

What is Named Entity Recognition?

Jim bought 300 shares of Acme Corp. in 2006.

Named Entity Recognition Classifier

PER   O   NUM   O   O   ORG   ORG   O   YEAR
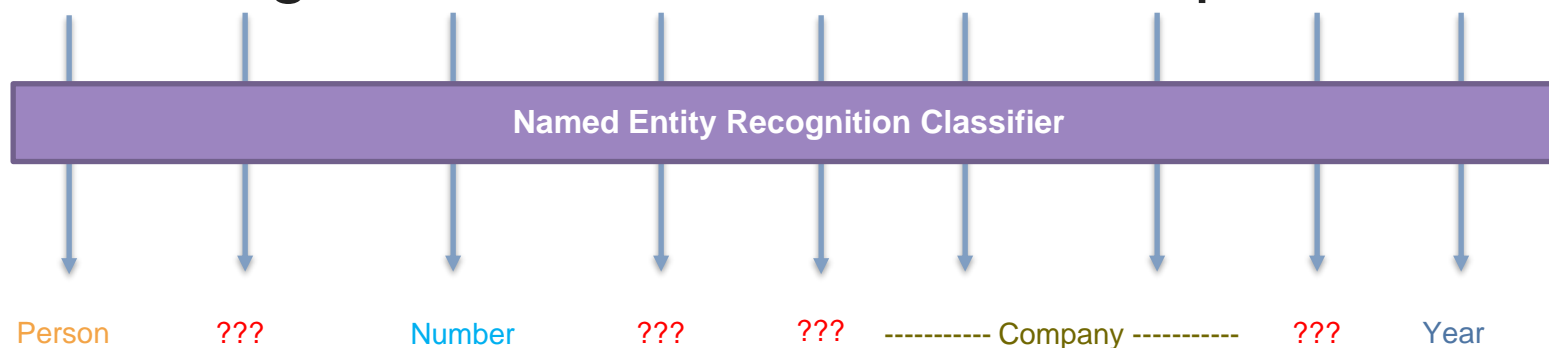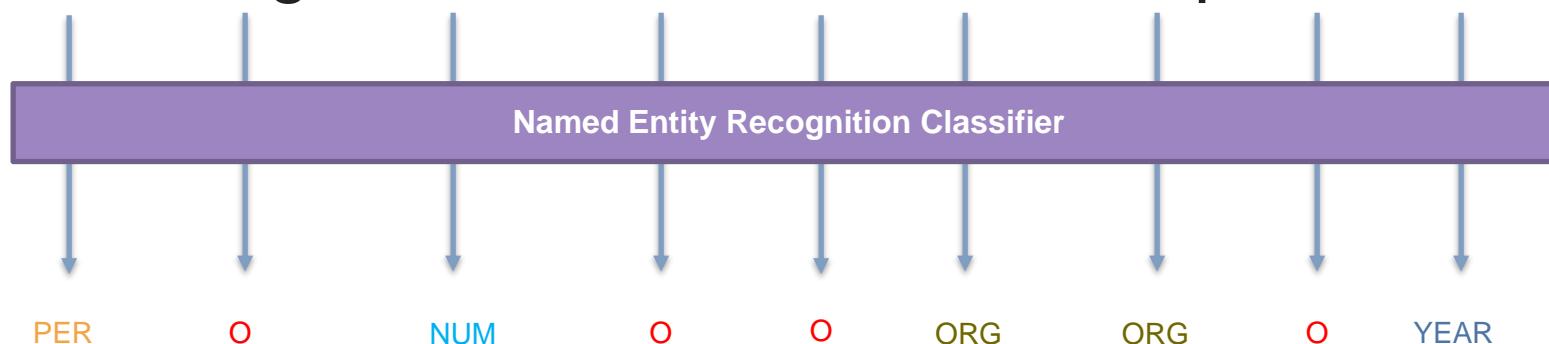
By convention, we give a code name for each entity we
want to recognize.

"O" is a special entity that we will want to ignore.

NER – Used for Extracting Actionable Data

Jim bought 300 shares of Acme Corp. in 2006.

Named Entity Recognition Classifier

B-PER   O   B-NUM   O   O   B-ORG   I-ORG   O   B-YEAR
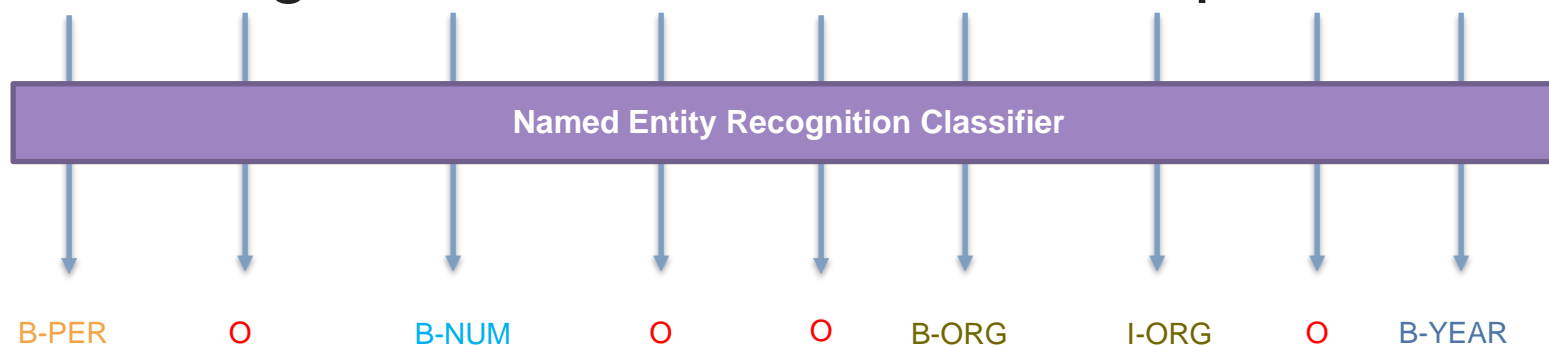
For multi-word entities like "Acme Corp", we use:
    B-ORG   (1st word), and
    I-ORG   (subsequent words joined to the first name)

This helps to extract important information that your app can act upon.

NER – Security and Personal Data Protection

Jim bought 300 shares of Acme Corp. in 2006.

Named Entity Recognition Classifier with Rules for Masking Entities (eg. People / Companies)

B-PER   O   B-NUM   O   O   B-ORG   I-ORG   O   B-YEAR
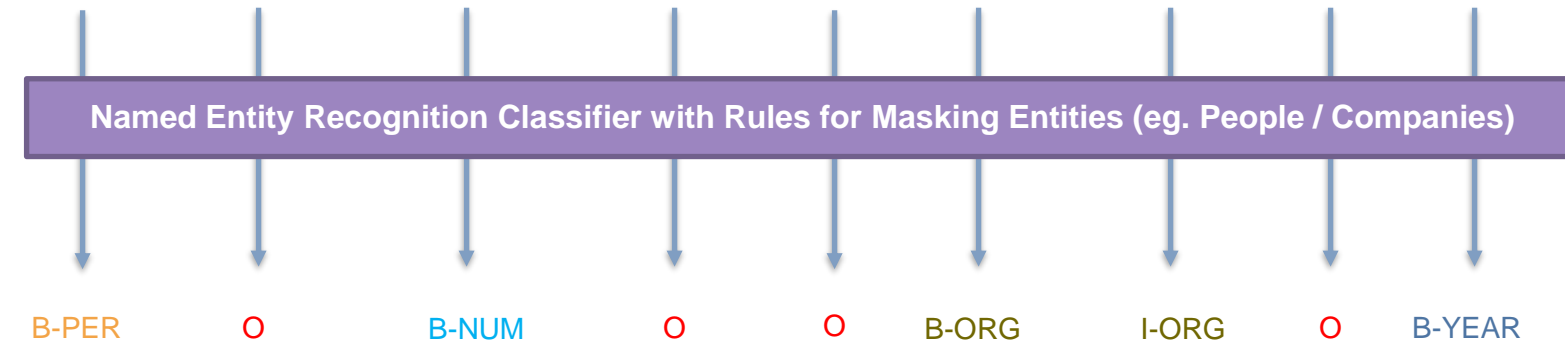
****** bought 300 shares of ****** ****** in 2006.

Your app can also use the classified labels for each word to selectively mask them.

Use Cases – Information Extraction

Masking sensitive data (names, gender, age) from medical history

Masking sensitive data (names, banking information) from emails / online chats

Extracting important information from medical history

https://youtu.be/6vX3Us1TOw8?t=159

Extracting actionable information from users in a chatbot conversation

Difference with Entity Recognition?

- Remember Entity Recognition in DialogFlow?

- Named Entity Recognition is a pre-defined list of entities learned by the AI model.

- DialogFlow's Entity Recognition allow you to define custom entity types. For example,
  I want to fix an appointment for repairing **bicycle wheels**

Custom entity with other possible values:
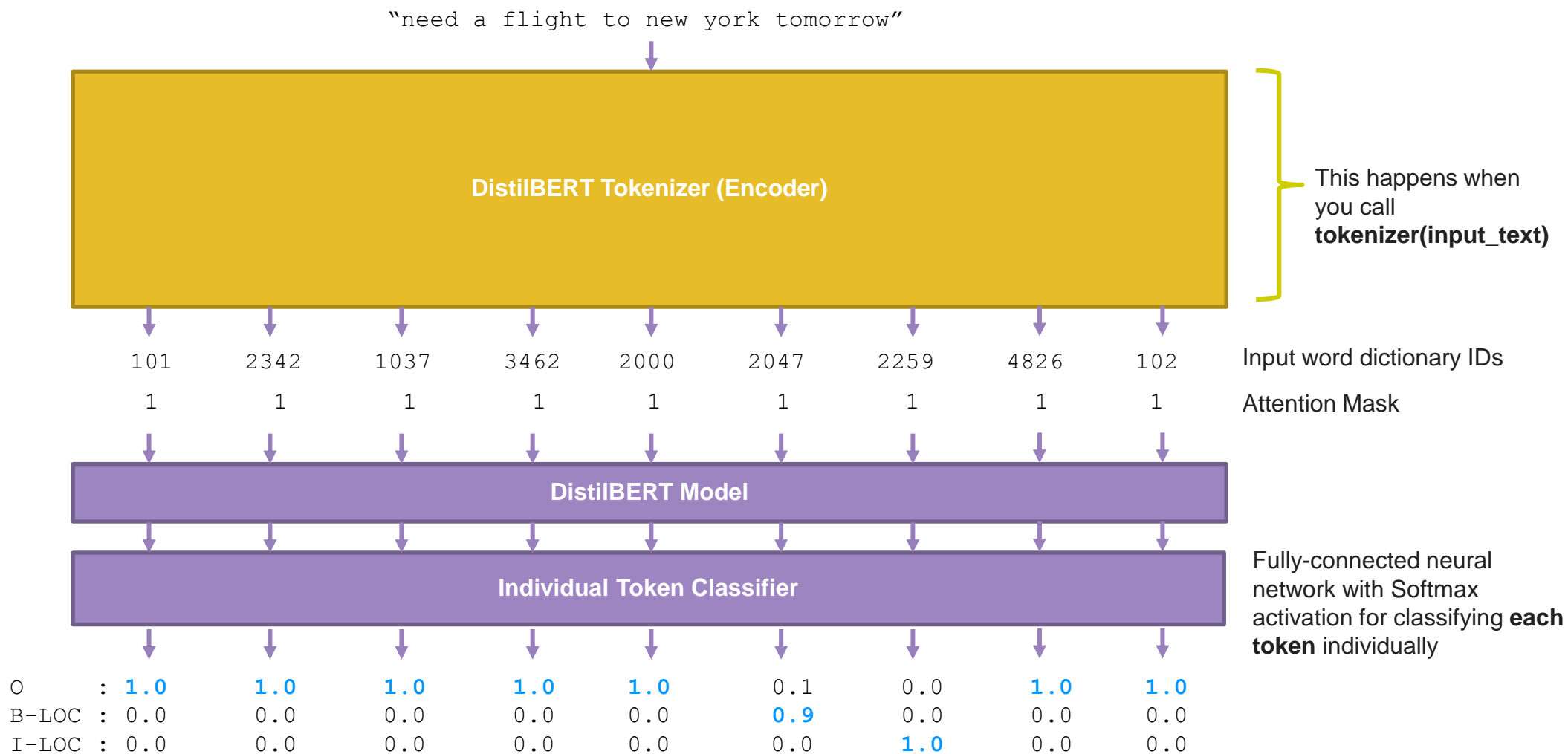Handlebar, speedometer, bike frame, etc

Transformers with HuggingFace Again

"need a flight to new york tomorrow"

DistilBERT Tokenizer (Encoder)

This happens when you call **tokenizer(input_text)**

101	2342	1037	3462	2000	2047	2259	4826	102	Input word dictionary IDs
1	1	1	1	1	1	1	1	1	Attention Mask

DistilBERT Model

Individual Token Classifier

Fully-connected neural network with Softmax activation for classifying **each token** individually

O :	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	0.1	0.0	**1.0**	**1.0**
B-LOC :	0.0	0.0	0.0	0.0	0.0	**0.9**	0.0	0.0	0.0
I-LOC :	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0

HuggingFace… Again?

- Model creation:
  - Since this is used for classifying individual tokens (words) to try to recognize whether they are names of places: we must create a new, separate model.

The I- O- B- Format

- As mentioned, to label our entities, we use IOB format.
- Also known as **Inside-Outside-Beginning:**

Jim	B-PER	This single word "Jim" forms the name of a person
bought	O	
2	O	
tickets	O	
to	O	The two words, "New York", form an instance of the location.
New	B-LOC	
York	I-LOC	The first word of an entity is marked with B-###, second word onwards marked with I-###

Training and Test Dataset

- This is an example of the CoNLL 2003 dataset:
  - https://github.com/synalp/NER/blob/master/corpus/CoNLL-2003/eng.testa

```
-DOCSTART- -X- O O

CRICKET NNP I-NP O
- : O O
LEICESTERSHIRE NNP I-NP I-ORG
TAKE NNP I-NP O
OVER IN I-PP O
AT NNP I-NP O
TOP NNP I-NP O
AFTER NNP I-NP O
INNINGS NNP I-NP O
VICTORY NN I-NP O
. . O O

LONDON NNP I-NP I-LOC
1996-08-30 CD I-NP O
```

5 Entity Types:

```
- B-PER / I-PER       Person Name
- B-LOC / I-LOC       Country or Location Name
- B-ORG / I-ORG       Organization or Company Name
- B-MISC / I-MISC     Any Other Names
- O                   Other Words to be Ignored
```

Training and Test Data Format

- The format required for the training and evaluation:

- X (inputs):
  - Output from the tokenizer (input word IDs, and attention mask)

- Y (output):
  - A list of indexes to the individual word classification (training: *not* one-hot encoding, inference: one-hot encoding)

Inference

Original text

need   a   flight   to   new   york   tomorrow

[CLS]   [SEP]

DistilBERT Tokenizer (Encoder)

Input IDs [101,   2064,   1045,   2113,   2054,   7599,   2024,   2800,   102]

Data to be fed into model for inference

Attention Mask [1,   1,   1,   1,   1,   1,   1,   1,   1]

DistilBERT Model + Token Classifier

O :	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	0.1	0.0	**1.0**	**1.0**
B-LOC :	0.0	0.0	0.0	0.0	0.0	**0.9**	0.0	0.0	0.0
I-LOC :	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0

Output from the token classifier (softmax)

argmax

0	0	0	0	0	1	2	0	0
Label: **O**	**O**	**O**	**O**	**O**	**B-LOC**	**I-LOC**	**O**	**O**

Handling Sub-words

- The DistilBERT Transformers is trained on a dictionary of about 30,000+ words

- Not all words can exist in the dictionary, some words are represented with multiple tokens of sub-words.

- For example, the word
  - "transformer" is tokenized to 2 words:
  - "transform" + "##er"
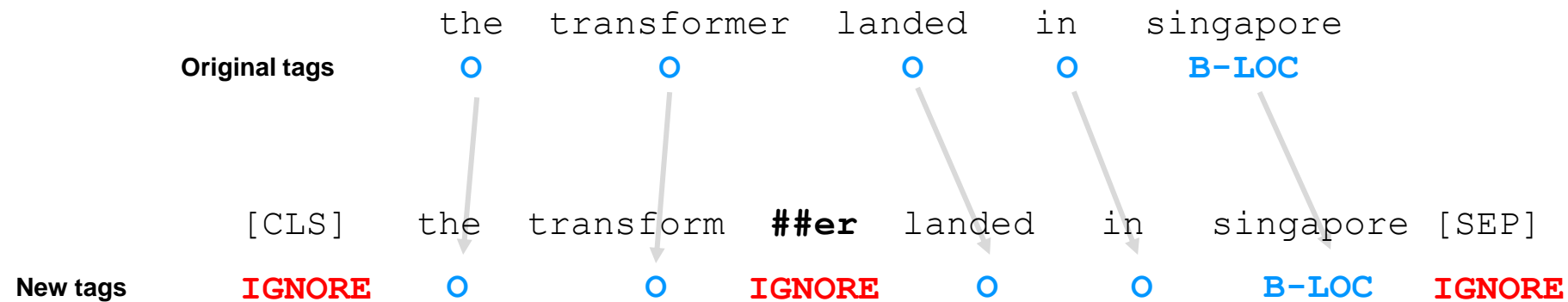
Handling Sub-words

- Big idea:
  - ignore the token labelling and classification for the sub-words.

the   transformer   landed   in   singapore

DistilBERT Tokenizer

[CLS]   the   transform   **##er**   landed   in   singapore [SEP]

Handling Sub-words

- Big idea:
  - ignore the token labelling and classification for the sub-words.

	the	transformer	landed	in	singapore
Original tags	O	O	O	O	**B-LOC**
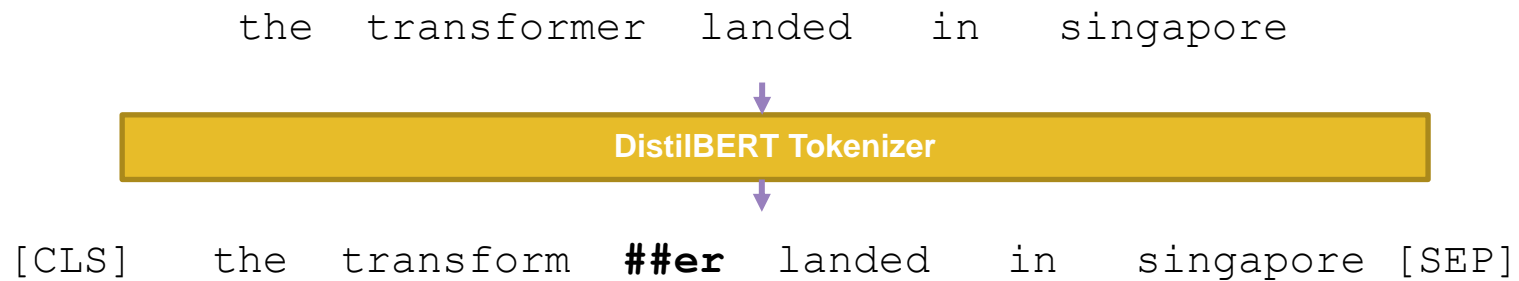
	[CLS]	the	transform	**##er**	landed	in	singapore	[SEP]
New tags	**IGNORE**	O	O	**IGNORE**	O	O	**B-LOC**	**IGNORE**

Inference

Original text

```
the transformer landed in singapore
```

Split into words

```
["the", "transformer", "landed", "in", "singapore"]
```

DistilBERT Tokenizer (Encoder with return_offset_mapping = True)

	[CLS]	the	transform	##er	landed	in	singapore	[SEP]
Input IDs	[101,	1996,	10938,	2121,	5565,	1999,	5264,	102]
Attention Mask	[1,	1,	1,	1,	1,	1,	1,	1]
Offset Mapping	[**(0,0),**	(0,3),	(0,9),	**(9,11),**	(0,6),	(0,2),	(0,9),	**(0,0)**]

Only Input IDs + Attention Mask are fed into model for inference

DistilBERT Model + Token Classifier

O :	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	0.0	**1.0**
B-LOC :	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0
I-LOC :	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Output from the token classifier (softmax)

argmax

0	0	0	0	0	1	2	0

Creating Your Own Chatbot

PUTTING IT ALTOGETHER

Putting It Altogether

User enters text

Intent Recognition

Named Entity Recognition

Instructions (Logic) to Craft Reply

System displays reply to user
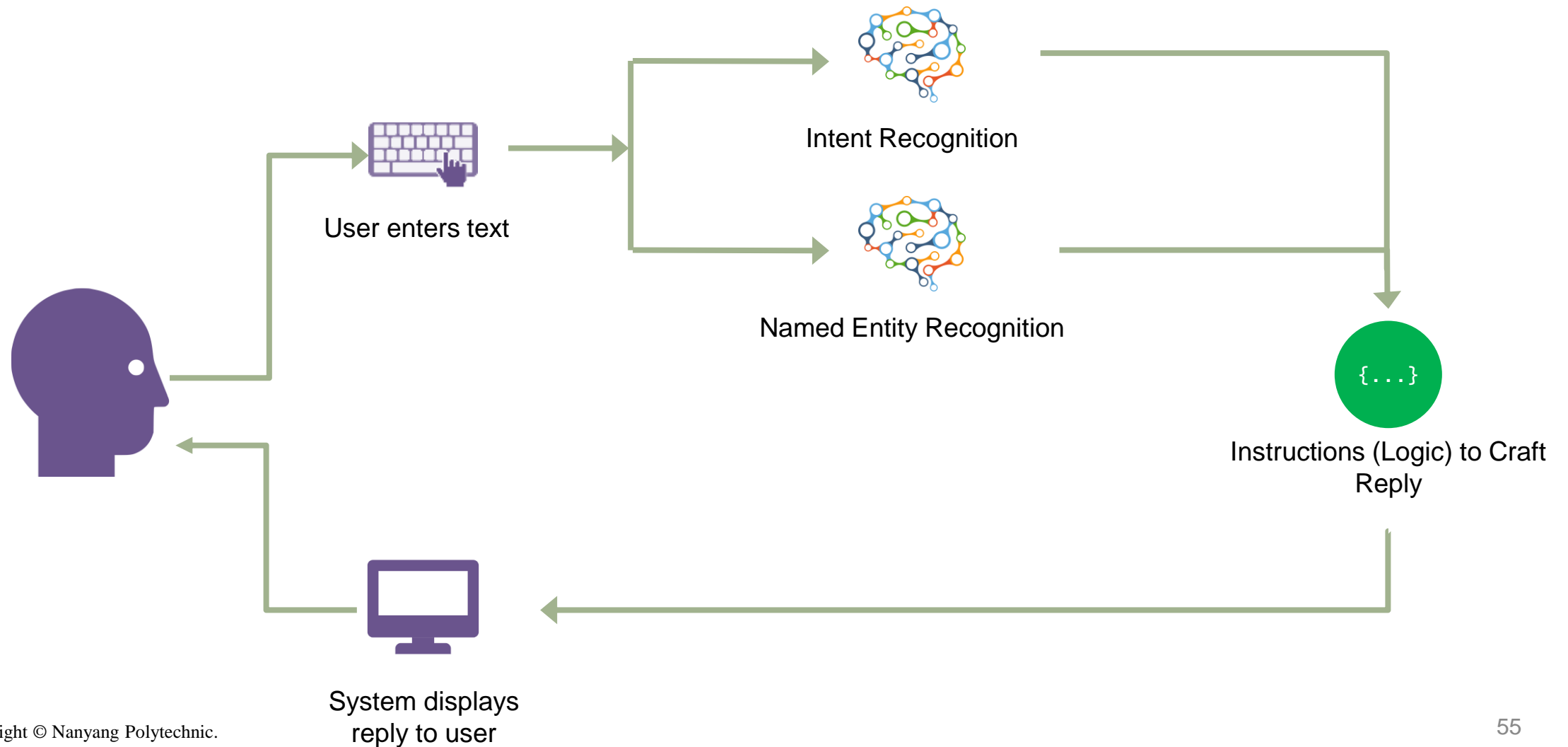
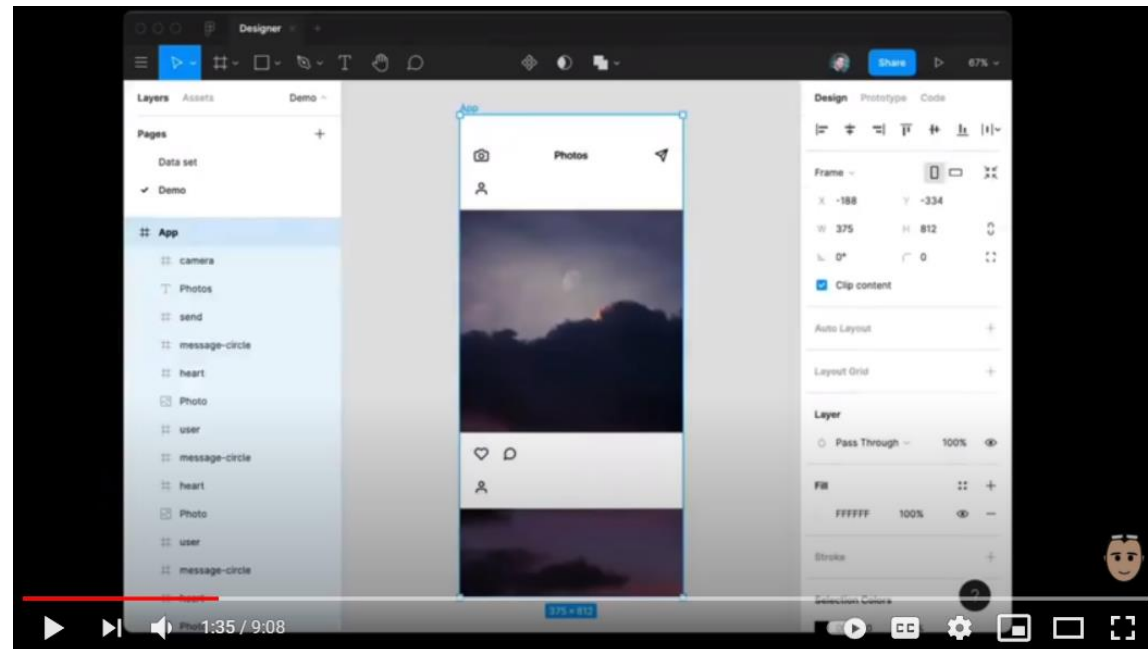**Creating Your Own Chatbot**

OTHER ADVANCES IN NATURAL LANGUAGE PROCESSING

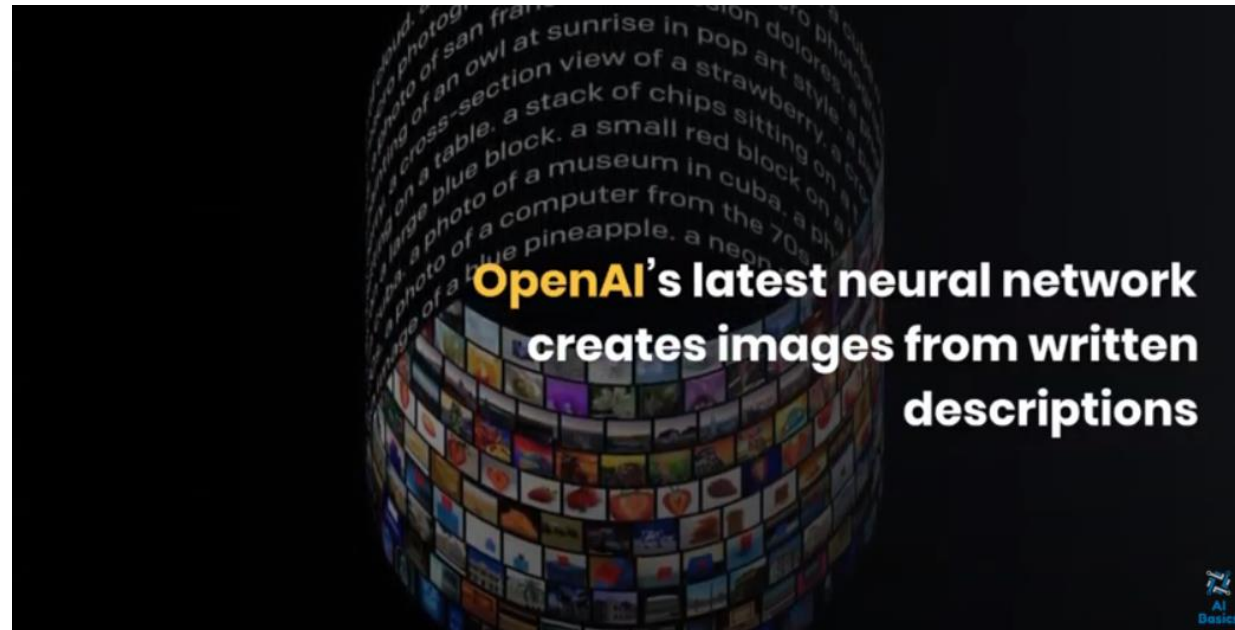# Other Advances in Natural Language Processing

OpenAI's GPT-3
https://en.wikipedia.org/wiki/GPT-3

https://www.youtube.com/watch?v=8psgEDhT1MM

Other Advances in Natural Language Processing

DALL.E (based on GPT-3)

https://daleonai.com/dalle-5-mins

https://www.youtube.com/watch?v=8psgEDhT1MM

Summary

- **What is a Chatbot?**
- **Transformers with the HuggingFace Library**
- **Intent Recognition**
- **Custom Named Entity Recognition**
- **Putting It Altogether**
- **Other Advances in Natural Language Processing**