B.Y.O.(C).B

Build Your Own (Chat) Bots

About me

- Chief Data Scientist @ Boost AI
- Machine learning enthusiast
- Kaggle junkie (highest world rank #3)
- Interested in:
  - Automatic machine learning
  - Large scale classification of text data
  - Chatbots

I like big data
and
I cannot lie

What are chatbots?

- Chatter bot
- Simulate human behavior
- Natural language processing
- Virtual assistants

Examples of chatbots

- A.L.I.C.E
- ELIZA
- Cleverbot
- Jabberwacky
- James (boost.ai)

Chatbot wars

ALICE vs ELIZA

Hello

Hi there!

Please go on.

Narration continues...

Narration continues...

OK.

I'm not sure I understand you fully.

That's good information.

Tell me more about that.

OK I will tell them when I see 'em.

Julie Bot vs ALICE

Hello

Hello.

Hi there!

how r u

I am not sure if I can quantify the answer. There might be more than one.

insane

I always try my best.

Do you play any sports?

I like to play the Turing Game.

ya

https://www.botlibre.com/browse?browse-type=ChatWar

Types of chatbots

- AIML pattern and rule based
- Sequence to sequence
- Intent based

Pattern or Rule Based

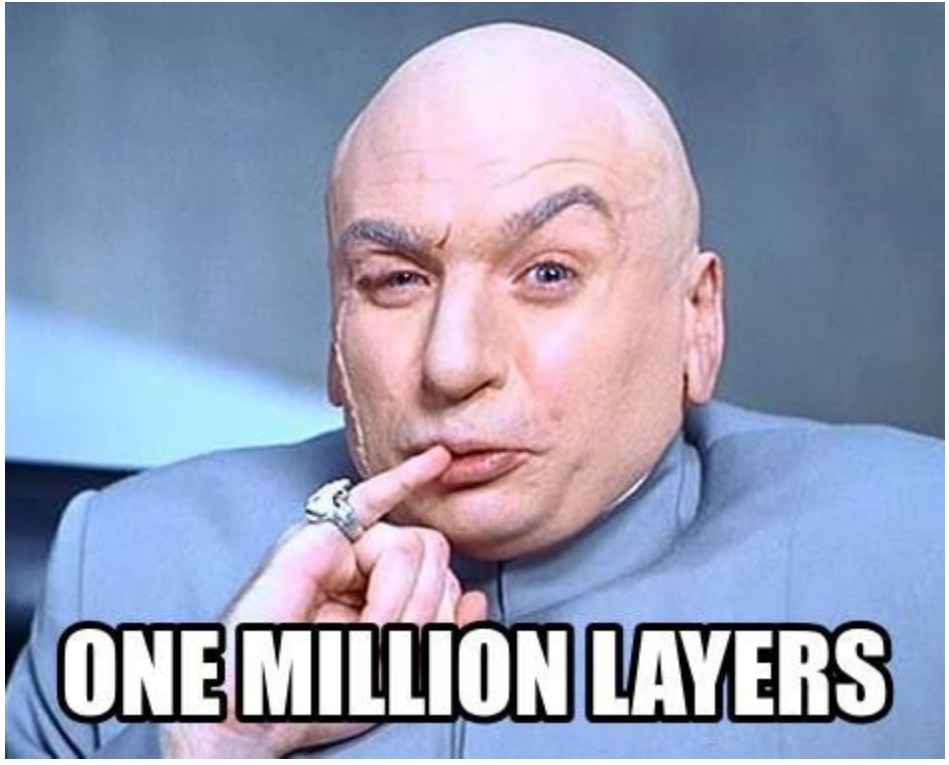
```
<aiml version="1.0.1" encoding="UTF-8">
  <category>
    <pattern>HELLO</pattern>
    <template>
            Well, hello!
        </template>
  </category>
  <category>
    <pattern>WHAT ARE YOU</pattern>
    <template>
            I'm a bot, silly!
        </template>
  </category>
</aiml>
```

Pattern or Rule Based

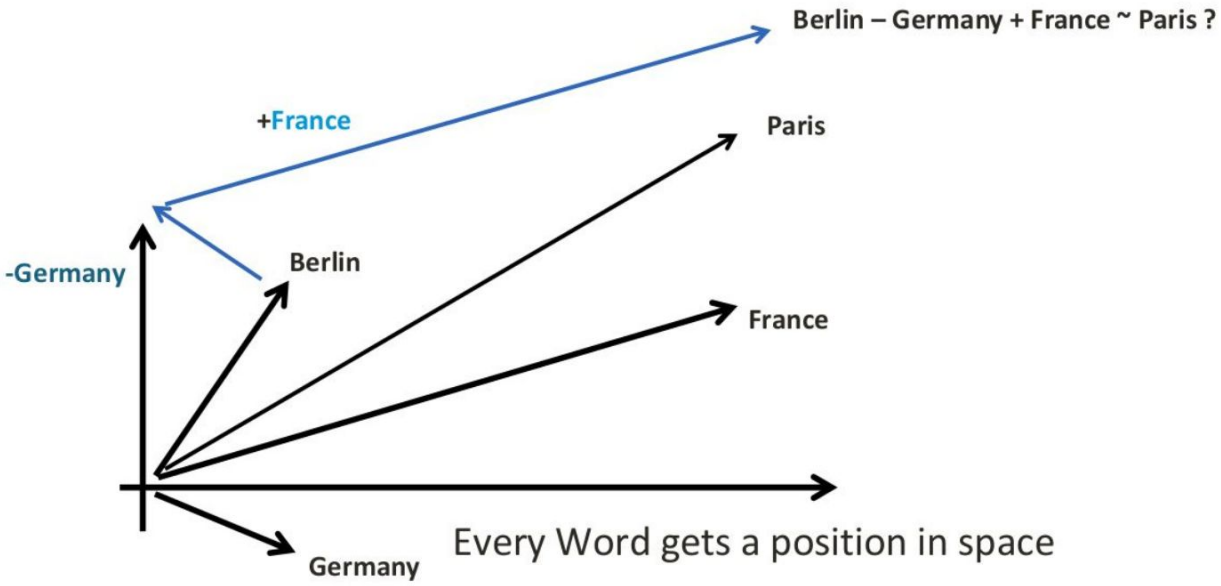
```
<category>
  <pattern>ONE TIME I *</pattern>
  <template>
    <random>
      <li>Go on.</li>
      <li>How old are you?</li>
      <li>Be more specific.</li>
      <li>I did not know that.</li>
      <li>Are you telling the truth?</li>
      <li>I don't know what that means.</li>
      <li>Try to tell me that another way.</li>
      <li>Are you talking about an animal, vegetable or mineral?</li>
      <li>What is it?</li>
    </random>
  </template>
</category
```

http://www.devdungeon.com/content/ai-chat-bot-python-aiml

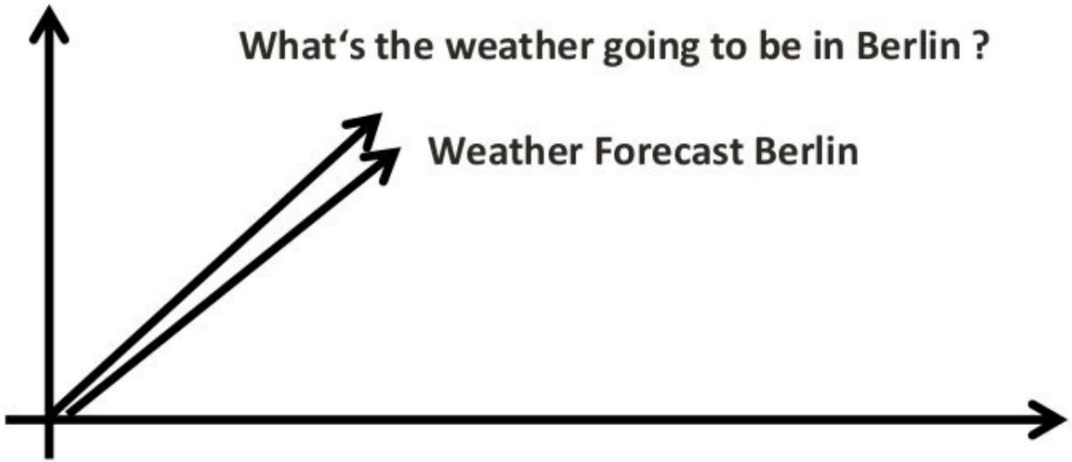
Deep Learning



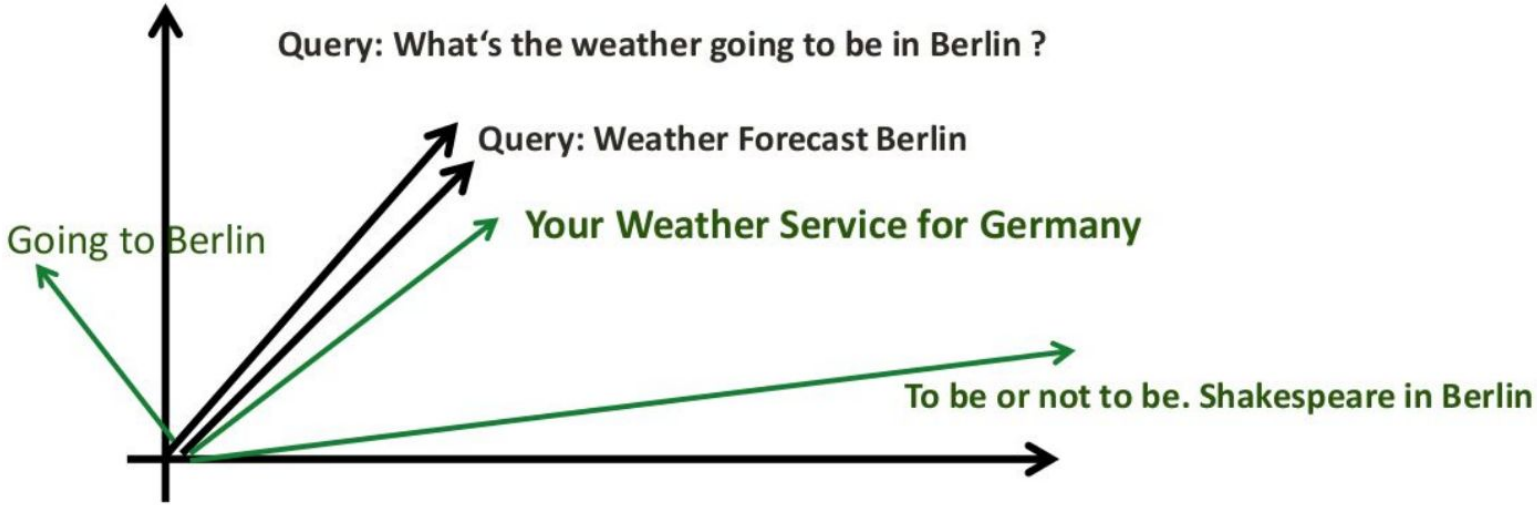
Word2Vec


Berlin – Germany + France ~ Paris ?
+France
Paris
-Germany
Berlin
France
Germany
Every Word gets a position in space

Word2Vec


What‘s the weather going to be in Berlin ?
Weather Forecast Berlin

Word2Vec


Query: What's the weather going to be in Berlin ?
Query: Weather Forecast Berlin
Your Weather Service for Germany
Going to Berlin
To be or not to be. Shakespeare in Berlin

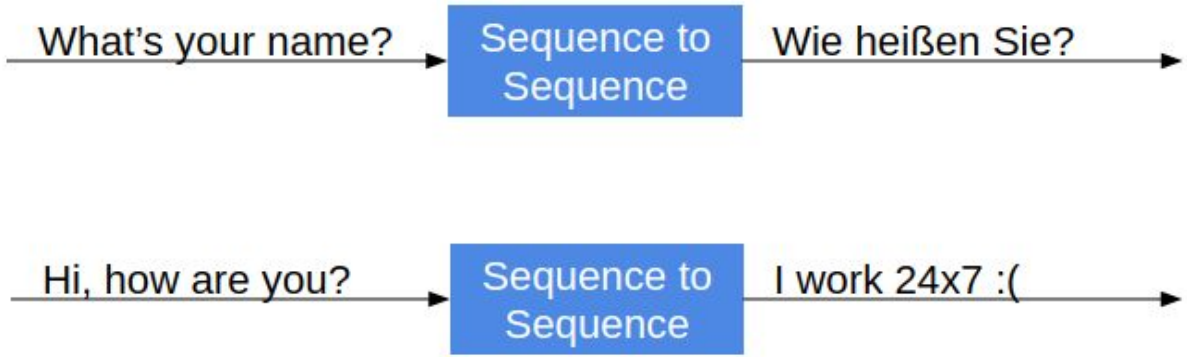
GloVe Embeddings

- Count based model
- Dimensionality reduction on co-occurrence counts matrix
- word-context matrix -> word-feature matrix
- Common Crawl
  - 840B tokens, 2.2M vocab, 300d vectors

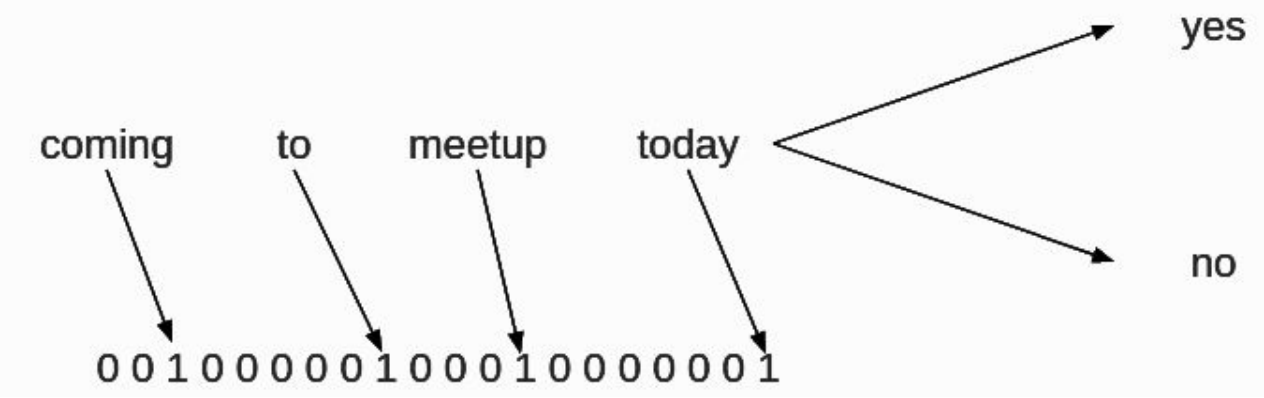
Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation

Sequence to Sequence

- Convert sequence from one domain to another
  - English <> German
  - Question <> Answer

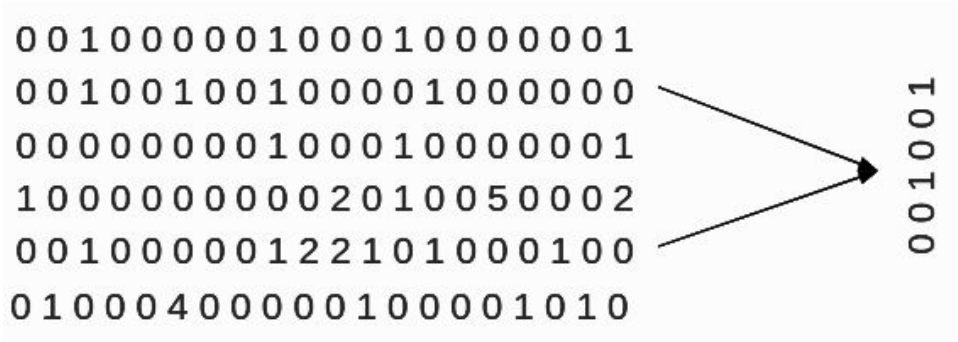


Sequence to Sequence



N - dimensions

Sequence to Sequence



Counts of the words in sentences

Target Variables: yes or no

TF-IDF

- TF(t) = Number of times a term t appears in a document / Total number of terms in the document
- IDF(t) = log(Total number of documents / Number of documents with term t in it)
- TF-IDF(t) = TF(t) * IDF(t)

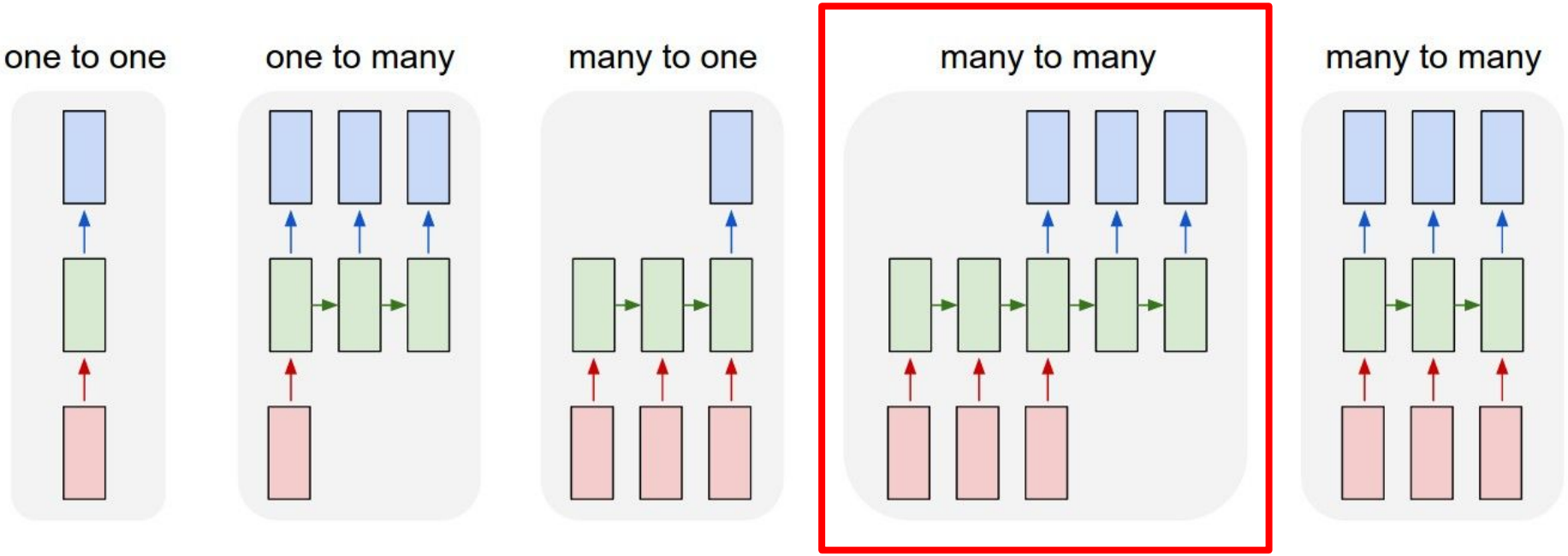
```
tfidf = TfidfVectorizer(min_df=3,  max_features=None,
            strip_accents='unicode', analyzer='word',token_pattern=r'\w{1,}',
            ngram_range=(1, 2), use_idf=1, smooth_idf=1, sublinear_tf=1,
            stop_words = 'english')
```

SVD

- Latent semantic analysis
- scikit-learn version of SVD
- 120 components

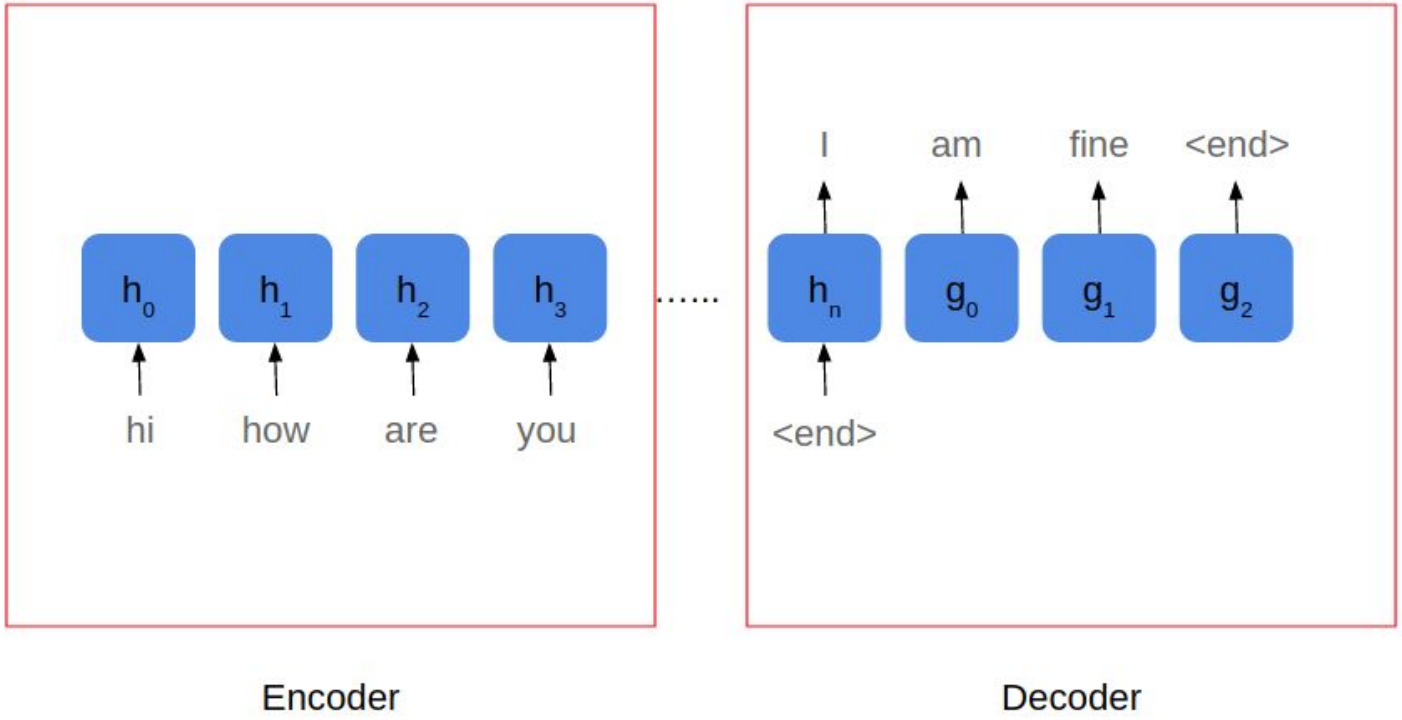
```
svd = decomposition.TruncatedSVD(n_components=120)
xtrain_svd = svd.fit_transform(xtrain)
xtest_svd = svd.transform(xtest)
```

Sequence to Sequence

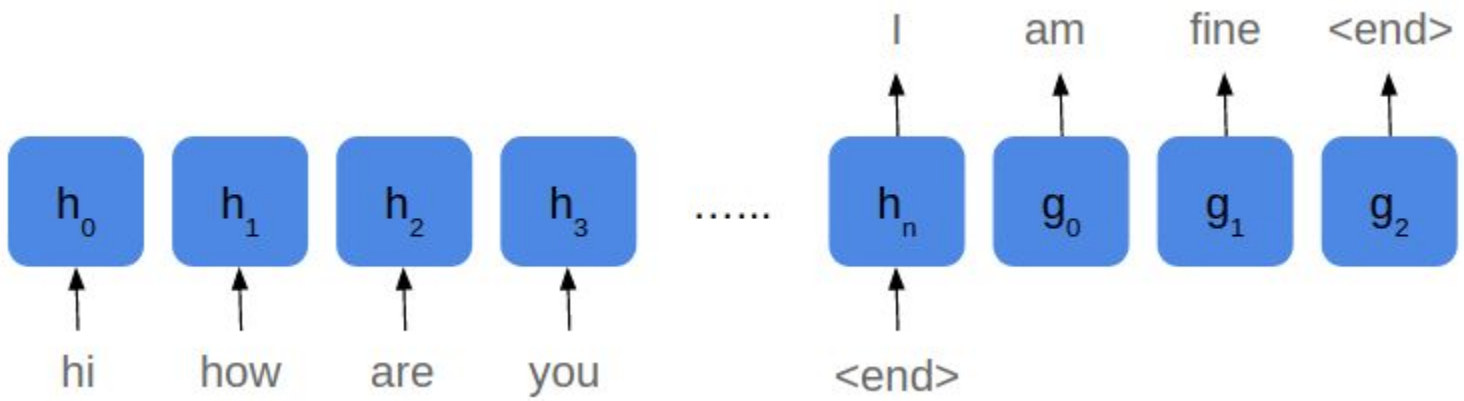


http://karpathy.github.io/2015/05/21/rnn-effectiveness/

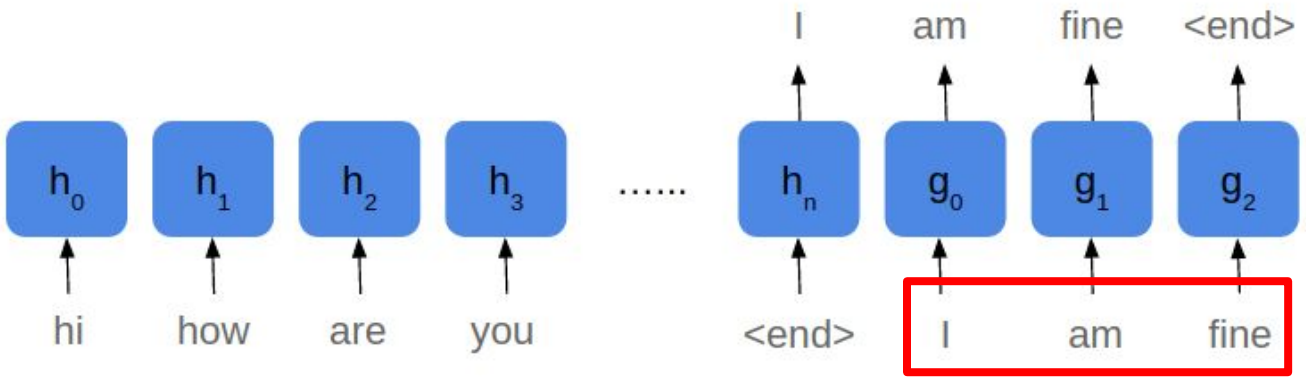
Sequence to Sequence



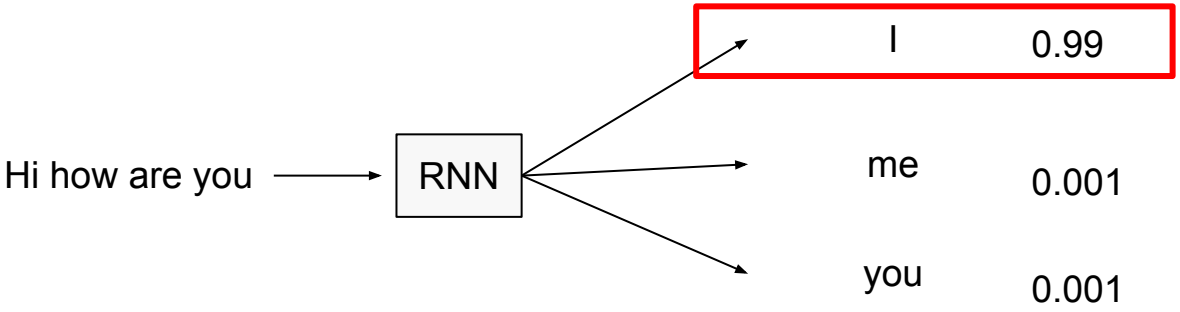
Sequence to Sequence



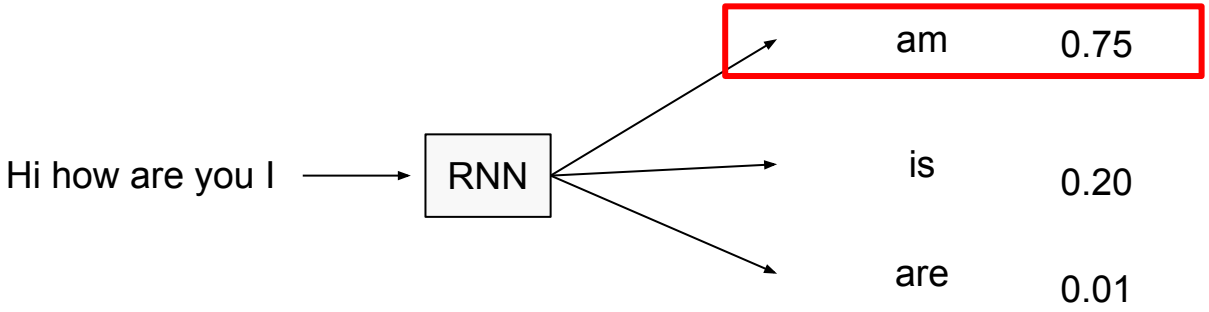
Sequence to Sequence



Predictions using Seq2Seq

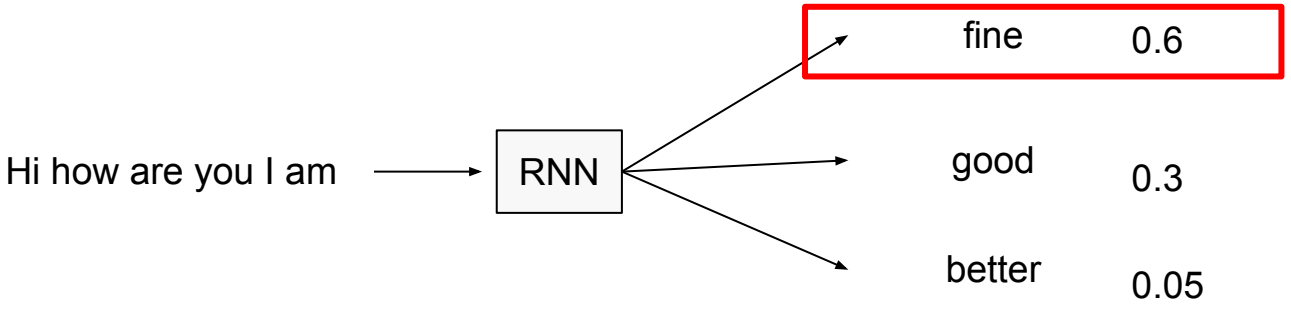


Predictions using Seq2Seq

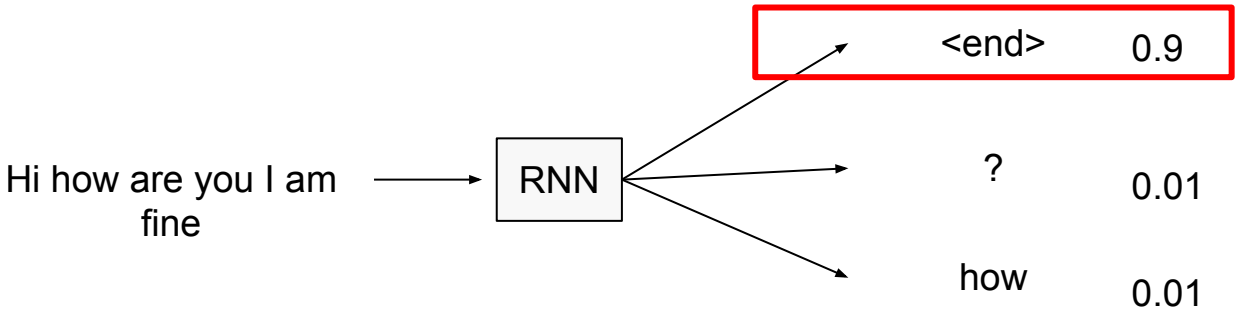


Top 3 Predictions

Predictions using Seq2Seq

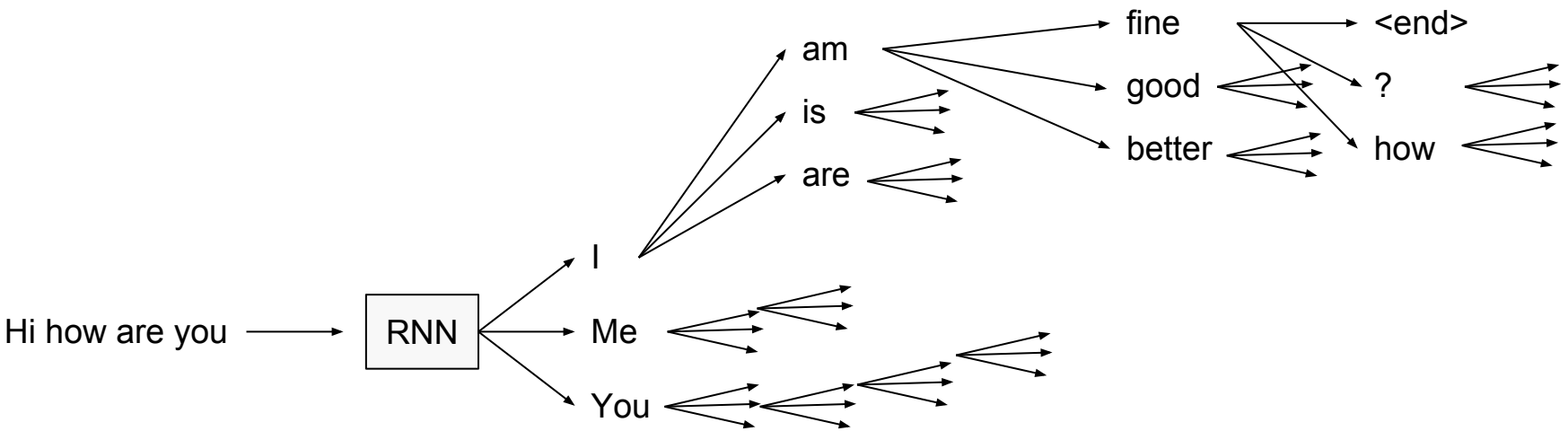


Predictions using Seq2Seq



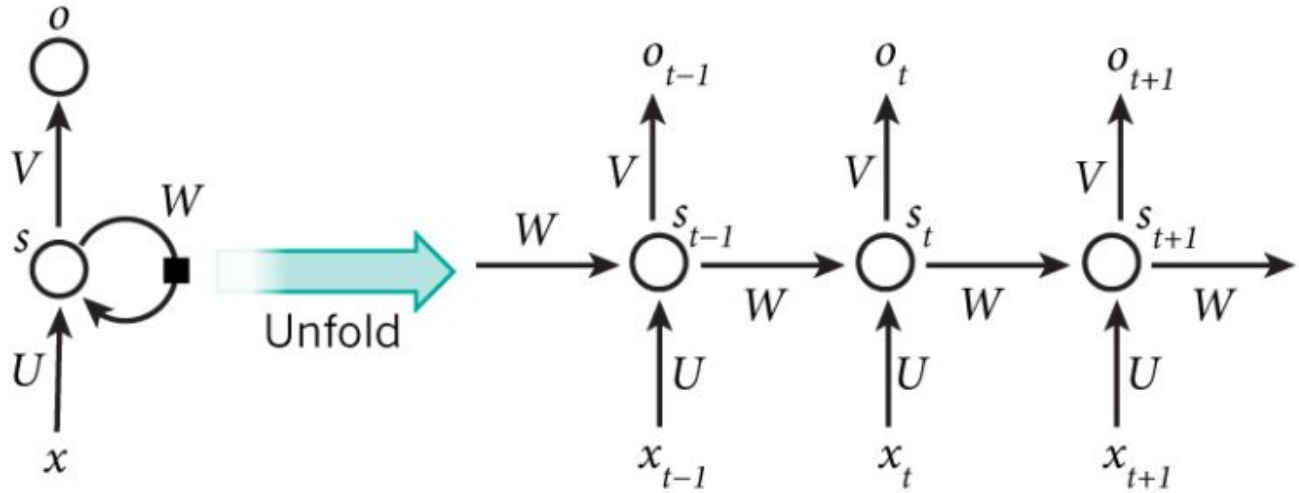
Top 3 Predictions

Predictions using Seq2Seq



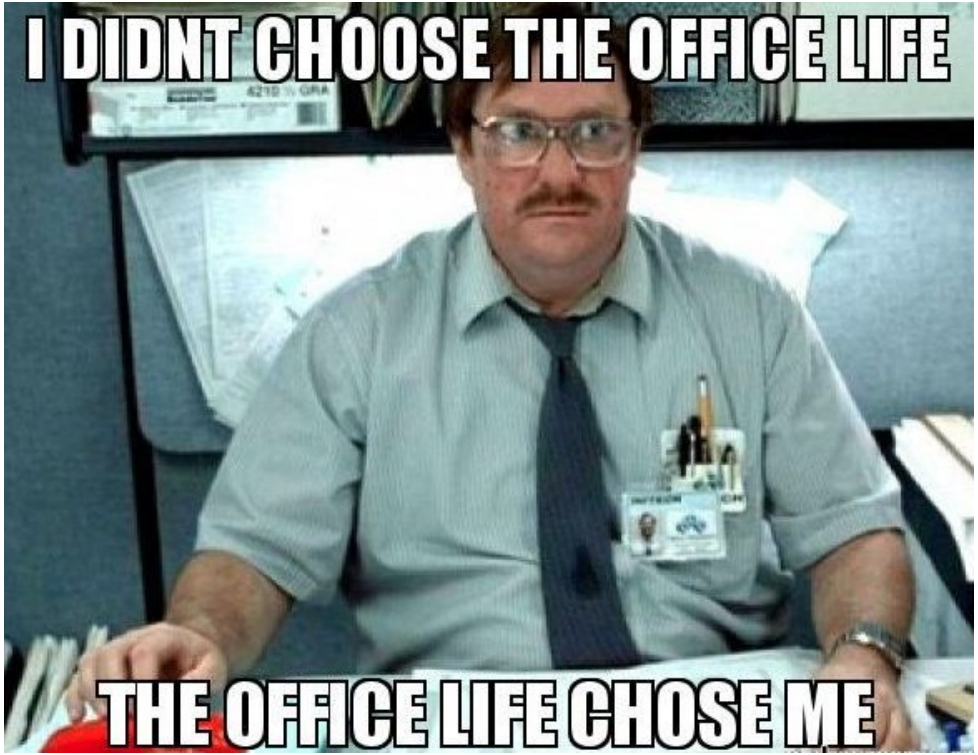
But what are RNNs?

- Sequential information
- Output dependent on previous input



A recurrent neural network and the unfolding in time of the computation involved in its forward computation. Source: Nature

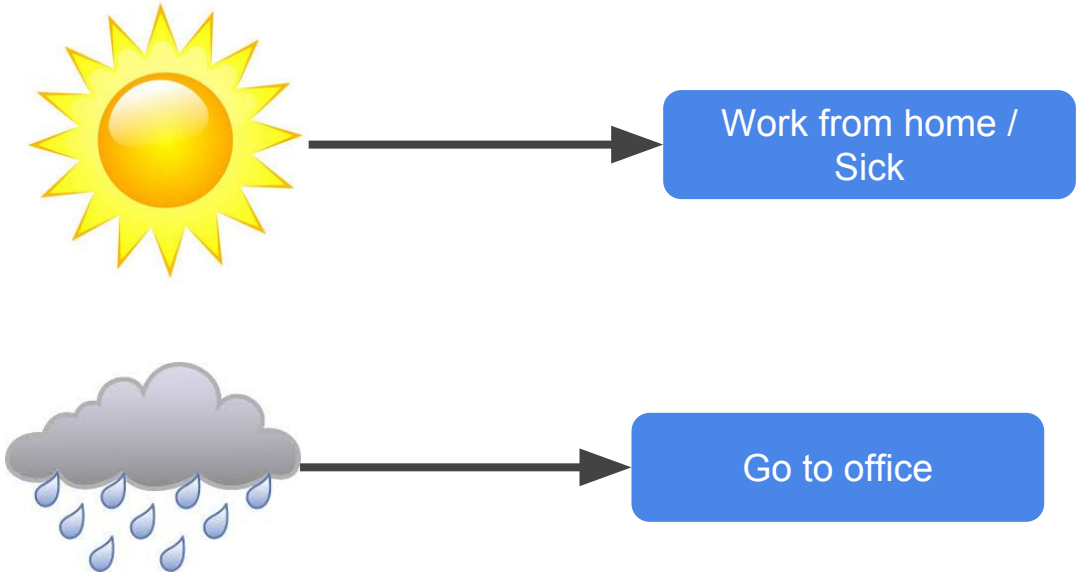
But what are RNNs?



But what are RNNs?

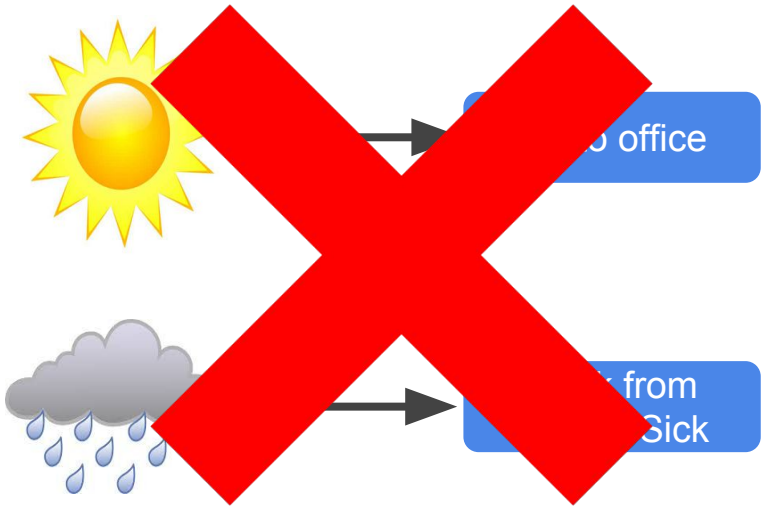
But what are RNNs?

But what are RNNs?

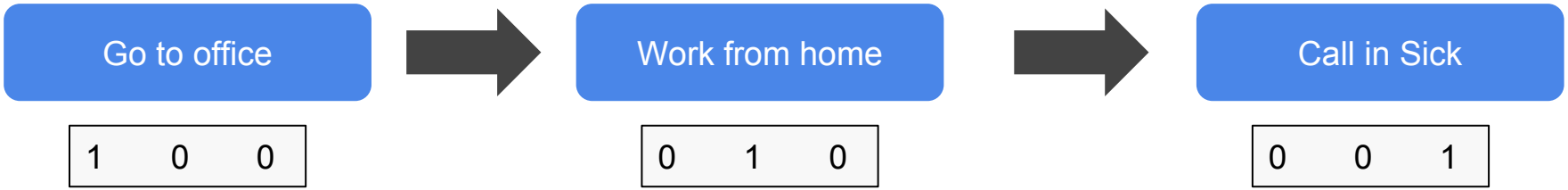


- A simple neural network depending on the weather.
- If it's sunny, I work from home.
- Bad weather implies going to the office.

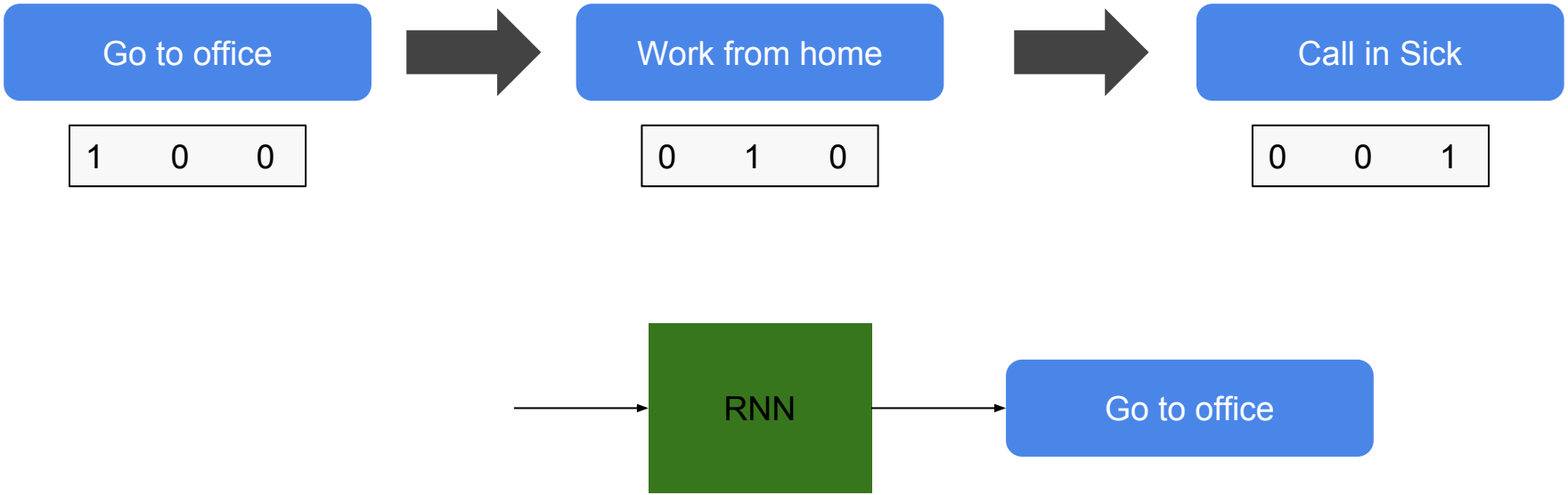
But what are RNNs?



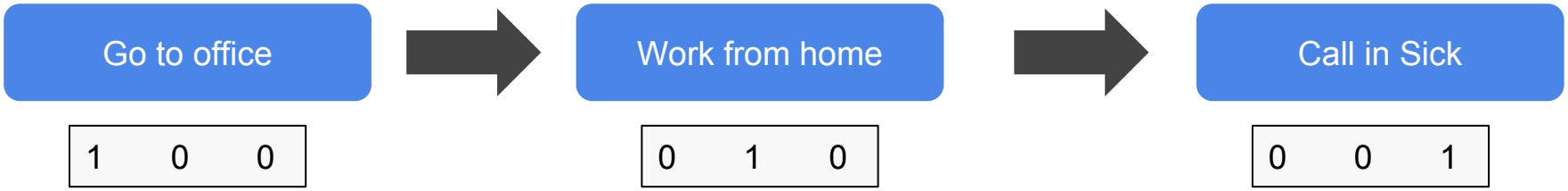
But what are RNNs?

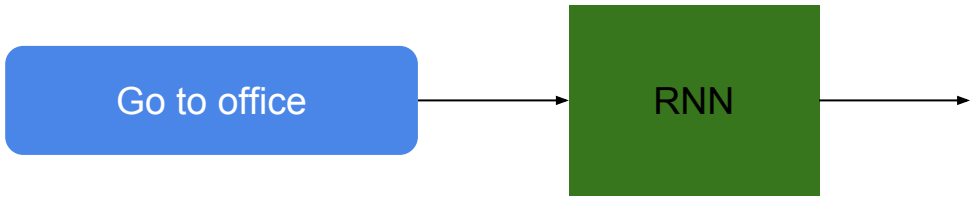


But what are RNNs?

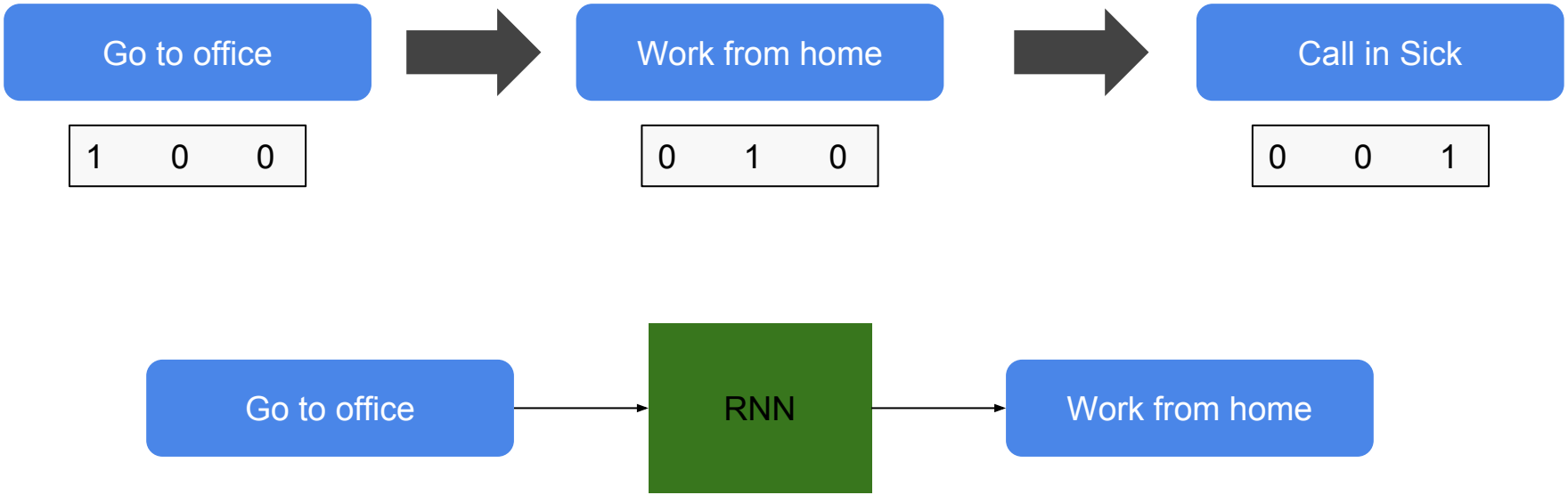


But what are RNNs?

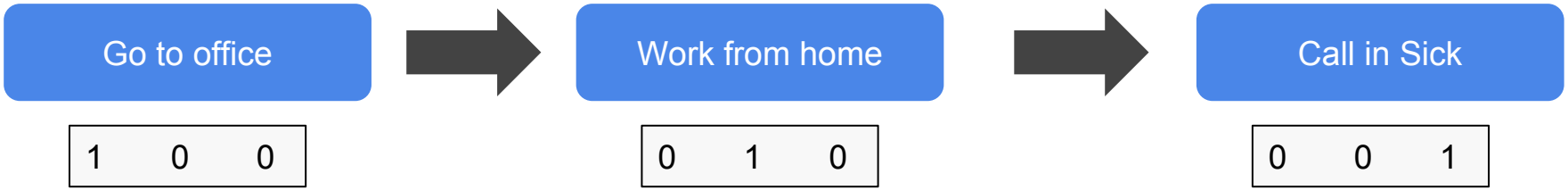


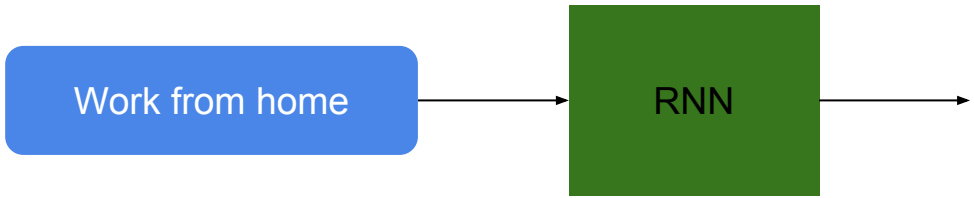


But what are RNNs?

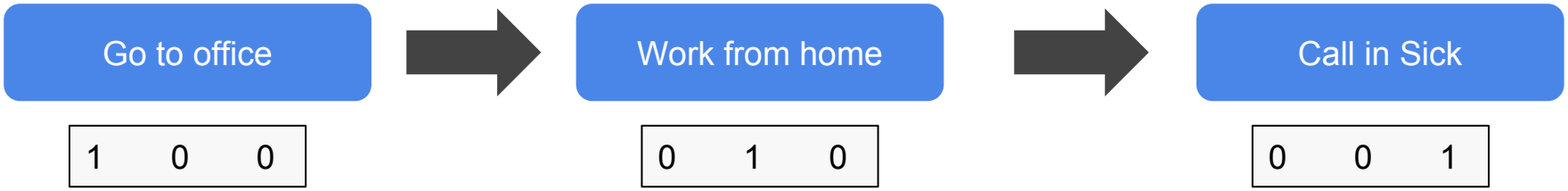


But what are RNNs?

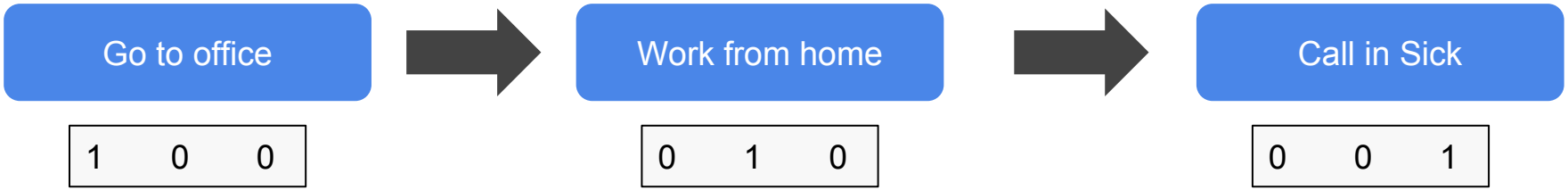


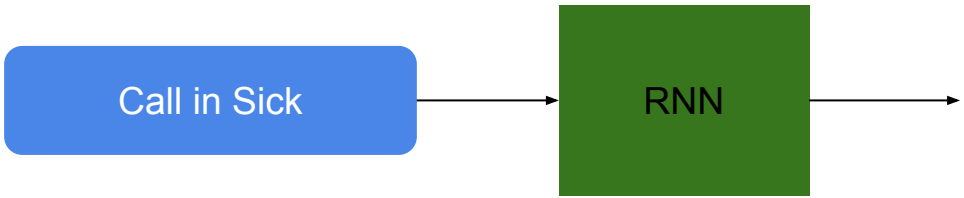


But what are RNNs?

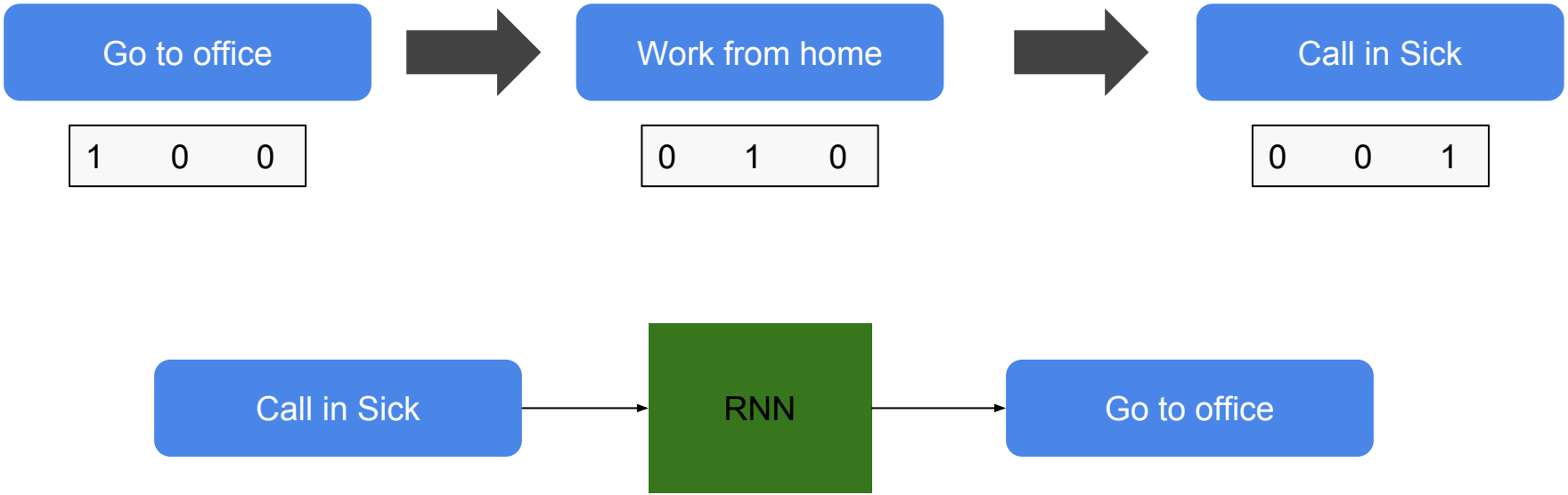


But what are RNNs?

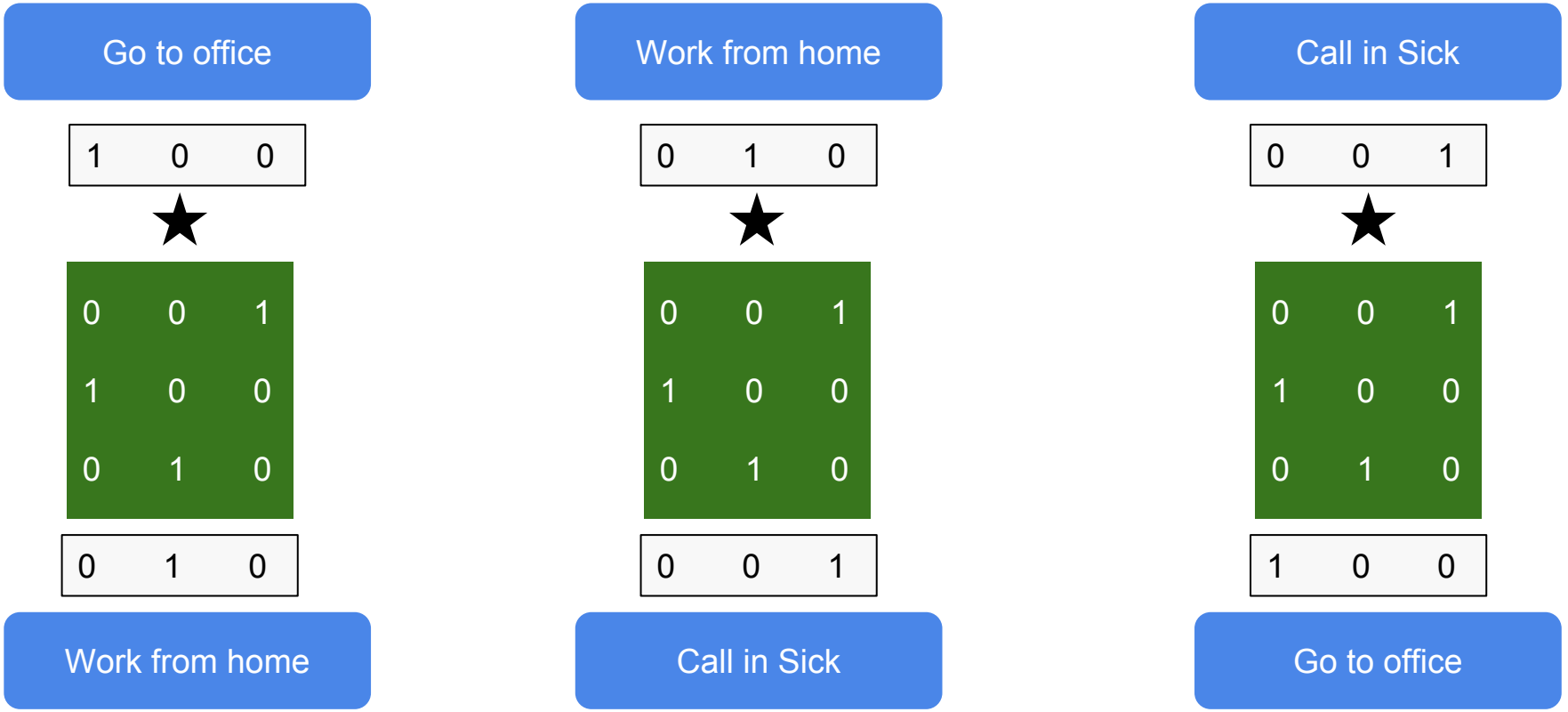




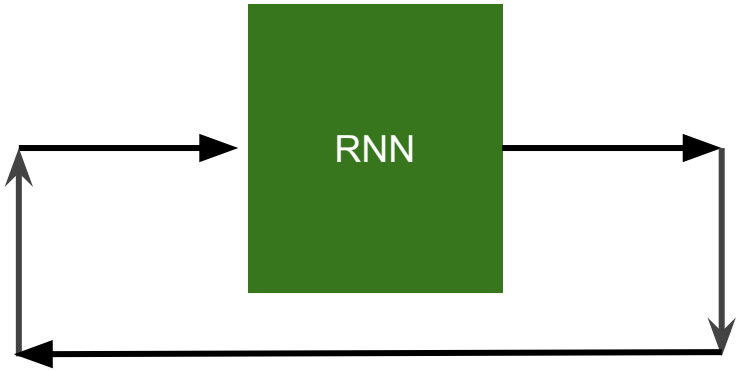
But what are RNNs?



But what are RNNs?



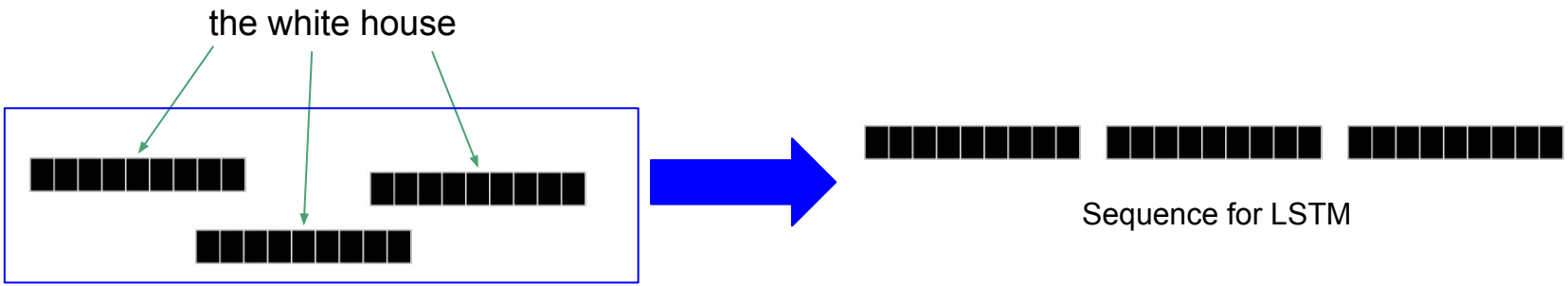
But what are RNNs?



LSTM

- Long short term memory
- A type of RNN
- Learn long term dependencies
- Used two LSTM layers

Feeding Data To LSTMs




```
model = Sequential()
model.add(LSTM(512, return_sequences=True, input_shape=(len(self.X[0]), len(self.X[0][0]))))
model.add(Dropout(0.2))
model.add(LSTM(128, return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(64, return_sequences=False))
model.add(Dropout(0.2))
model.add(Dense(1000,activity_regularizer=activity_l2(0.01)))
model.add(Dense(len(self.nd.categories),activity_regularizer=activity_l2(0.01)))
model.add(Activation('sigmoid'))

model.compile(loss='binary_crossentropy', optimizer='adam')
```

- United States
- President
- Politician
- Washington
- Lawyer
- Secretary

Embedding Layers

- Simple layer
- Converts indexes to vectors
- [[4], [20]] -> [[0.25, 0.1], [0.6, -0.2]]

1-D CNNs

- One dimensional convolutional layer
- Temporal convolution
- Simple to implement:

```
for i in range(sample_length):
    y[i] = 0
    for j in range(kernel_length):
        y[i] += x[i-j] * h[j]
```

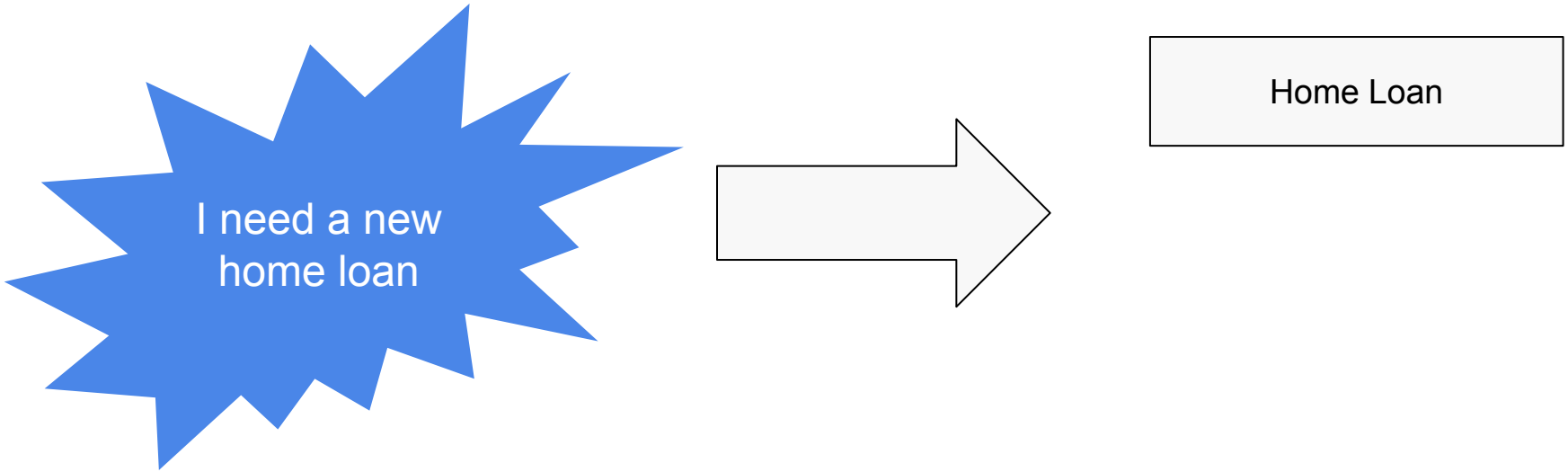
Training a Simple RNN

Hi.	Grüß Gott!
Run!	Lauf!
Wow!	Potzdonner!
Wow!	Donnerwetter!
Fire!	Feuer!
Help!	Hilfe!
Help!	Zu Hülf!
Stop!	Stopp!
Wait!	Warte!
Hello!	Hallo!
I try.	Ich probiere es.
I won!	Ich hab gewonnen!
I won!	Ich habe gewonnen!

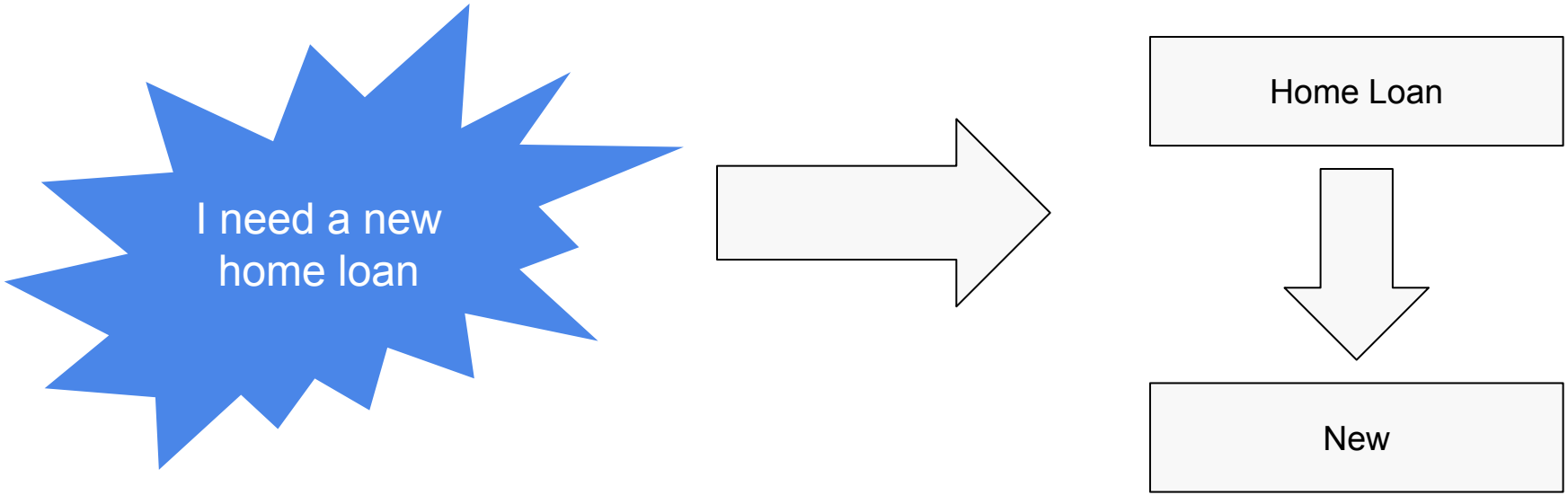
Intention Based Chatbots

- Every “chat” has an intent
- RNNs without seq2seq
- Multi class classification problem
- Isn’t as easy as it sounds!

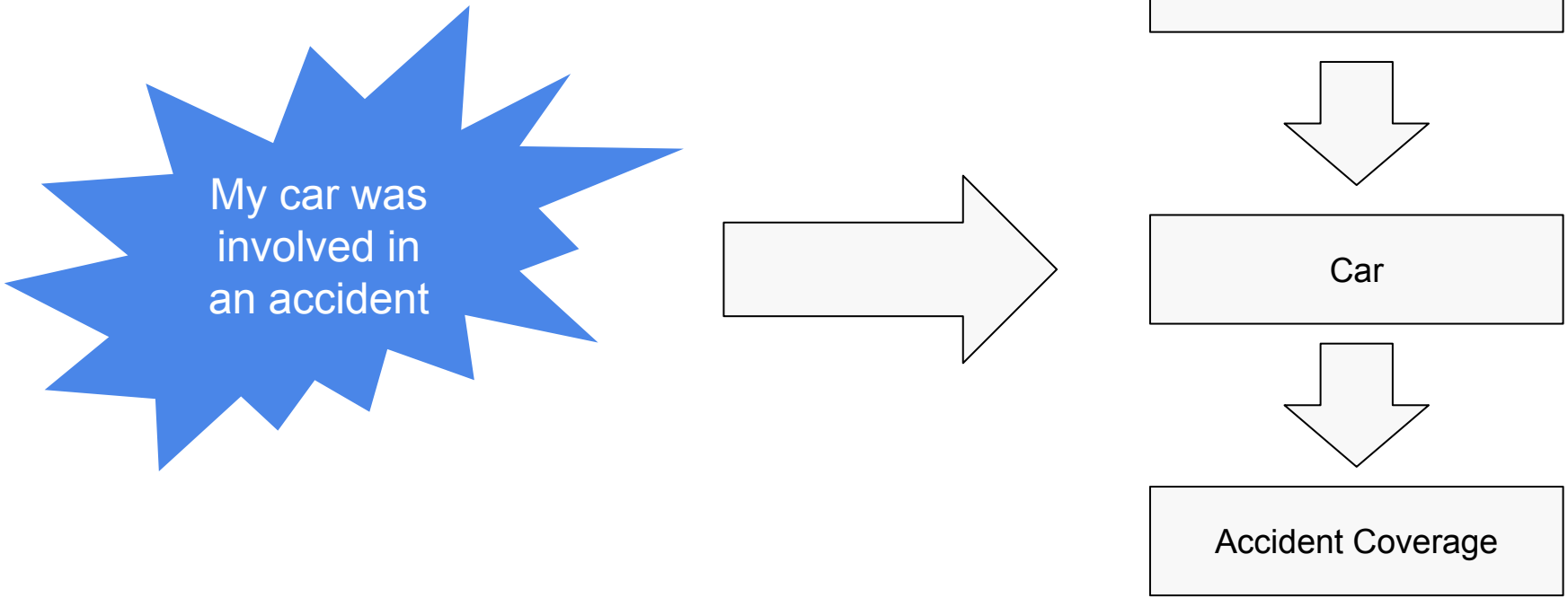
Intention Based Chatbots



Intention Based Chatbots



Intention Based Chatbots

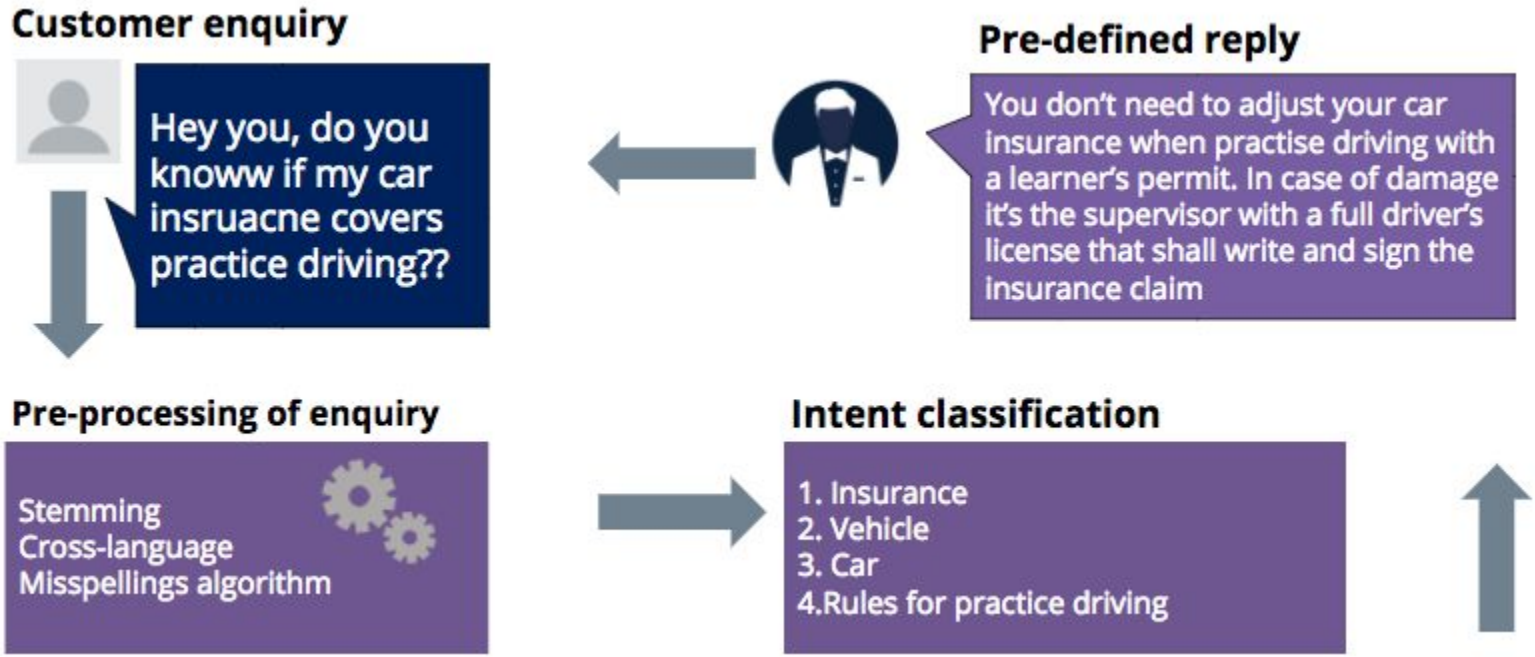


Intention Based Chatbots

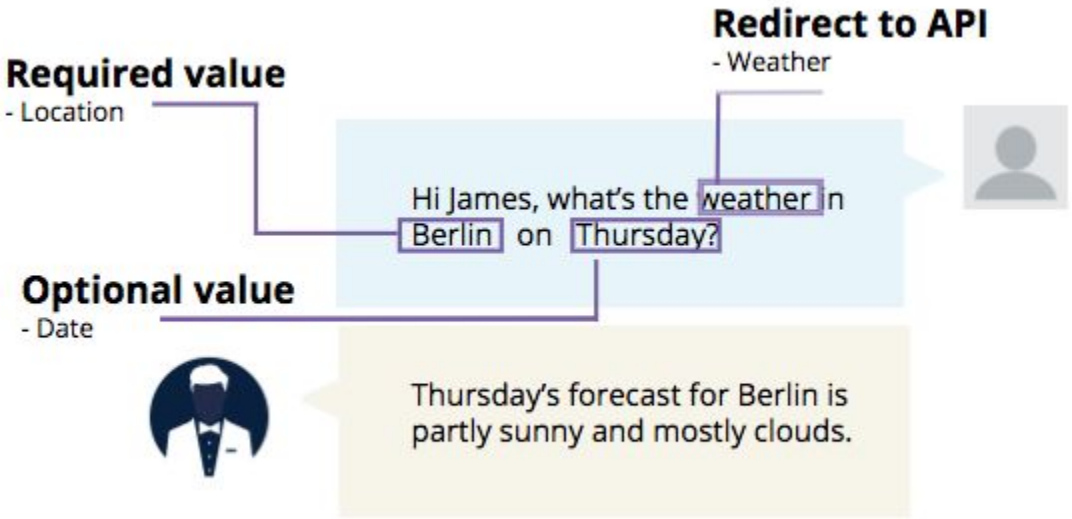
Credit Card Application

Child Insurance

Intention Based Chatbots



Intention Based Chatbots



Thank you

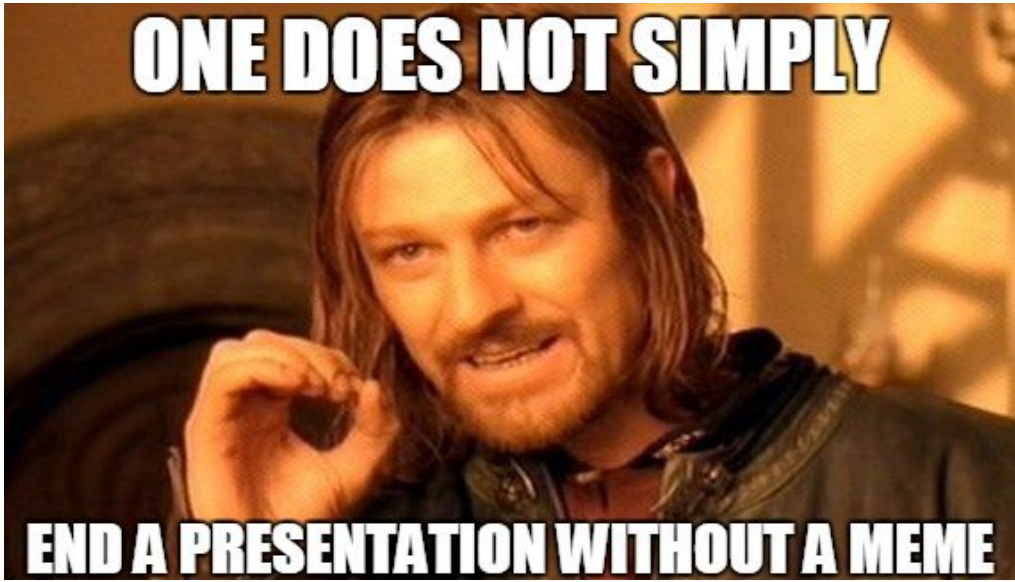
Questions / Comments?

A Lot of Code:

- ❖ github.com/abhishekkrthakur

Get in touch:

- ➢ E-mail: abhishek4@gmail.com
- ➢ LinkedIn: bit.ly/thakurabhishek
- ➢ Kaggle: kaggle.com/abhishek
- ➢ Twitter: @abhi1thakur



If everything fails, use Xgboost