6COSC020W.1 Applied AI Natural Language Processing

Dr Maria Chondrogianni

M.N.Chondrogianni@westminster.ac.uk

Introduction to Natural Language Processing

In this lecture we will:

- Define the field and discuss the state of the art.
- Highlight the industry needs for Natural Language Engineers
- Suggest why Natural Language Engineering provides a number of challenges for developers
- Offer a simple and robust solution for a Question-Answering system.

Introduction to Natural Language Processing

- Language has been at the centre of Artificial Intelligence research since its birth
- It offers a way to exhibit what human intelligence is all about
- It offers the opportunity to test the degree of *intelligence* of software/hardware (e.g. Turing test).

Introduction to Natural Language Processing

- Natural Language Processing
- Natural Language Generation
- Natural Language Engineering
- Speech Processing/ Speech Generation

Problems we try to solve

- Question- Answering
- Information Extraction
- Sentiment Analysis
- Machine Translation

The State of The Art
(slide based on Martin Jurafsky's 2018 video)

Areas where development solutions have mostly been provided:

- Spam detection (a relatively easy classification task)
- Part of speech tagging

ADJ ADJ NOUN VERB ADV
Colourless green ideas sleep furiously.

- Named entry recognition (NER)

PERSON ORG LOC
Einstein met with UN Officials in Princeton.

The State of The Art
(slide based on Martin Jurafsky's 2018 video)

Areas where we are making good progress in providing solutions:

- Sentiment analysis

e.g. Best roast chicken!

The waiter ignored us for 20 minutes.

- Coreference resolution

John told Peter he would travel to London.

The State of The Art
(slide based on Martin Jurafsky's 2018 video)

Areas where we are making good progress in providing solutions:

- Word Sense disambiguation (WSD)

I need new batteries for my mouse

- Parsing
- (Automatic) Machine Translation
- Information Extraction

e.g. You are invited to a party, Sunday 21st November, 7pm. (event, date, time).

The State of the Art

Areas where further research is needed:

- Question- Answering (QA)
- Paraphrasing

e.g. XYZ acquired ABC yesterday.

ABC has been taken over by XYZ.

- Summarisation
- Dialogue

Language complexity

- Why is language so difficult?

Language complexity

- Sounds (Phonetics and Phonology)
- Words (Morphology)
- Structure (Syntax)
- Meaning (Semantics)
- Use (Pragmatics)

Language complexity

- Non-standard English
- Segmentation issues (e.g. New York)
- Idioms (e.g. Break a leg!)
- Neologisms (e.g. ‘unlike’)
- World knowledge
- Entity names (e.g. films/records’ titles)

Tools that we need to be able to process (parse) or produce (generate) NL

- Knowledge about language
- Knowledge about the world
- A way to combine knowledge sources

Different ways to develop language systems

- Probabilistic models built from language data

e.g. P (‘maison’ → ‘house’) High

P (‘l’avocat general’ → ‘the general avocado’) Low

GPT-3 General Pre-trained Transformer 3

An autoregressive (i.e. a representation of a type of random process) language model which uses *deep learning* to produce human-like text.

- Information Retrieval models

A simple Dialogue system-
Dialogue management systems using Finite State Automata

Different types of dialogue management systems

- System-led applications

The system asks questions to elicit required parameters to task

- User-led applications

The user asks questions to obtain information

- Mixed-initiative applications

The user and the system elicit information, clarify unclear information, develop a common plan

Approaches to dialogue management

- Finite State methods

-dialogue structure: State Transition Network

-nodes= system's questions, network's paths= legal dialogues

-sub-dialogues in form of loops (supporting modular approach and libraries)

- Self-organising

-dynamically evolving structure

-based on computation of next dialogue act

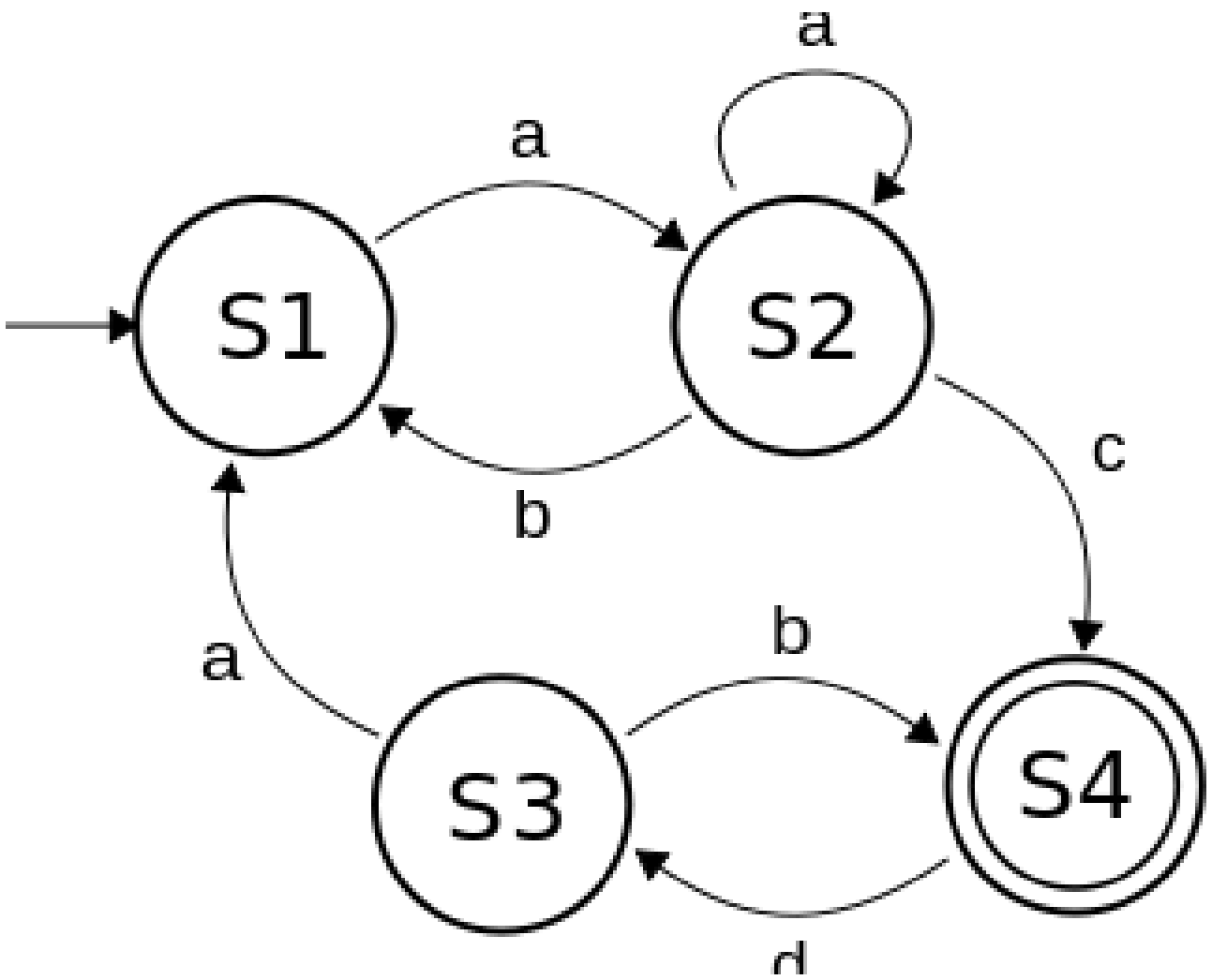
-several variants: plan-based, event-driven, frame based, among others.

Finite state machines

A simple way to **generate** natural language is through a **finite state automaton**.

This is a simple computing machine that outputs a sequence of symbols.

It starts from a start state and than tries to reach a final state, by making a **transition**, or a series of transitions, from one state to another.



Finite state machines

- A Finite State Automaton (FSA) has to keep moving states until it reaches a final state, where it can stop.
- We can write a simple program for a **laughing machine,** where when our generator moves from state 1 to state 2 it writes **h;** from state 2 to state 3 it writes **a;** and from state 3 it either goes back to state 1 by writing **nothing (#),** or it mover to state 4 by writing **!.** Our machine will be able to write **ha!** or **hahaha!** or **hahahaha!** *(output tape).*
- An FSA can be thought as a **directed graph.**

Finite State Machines- Recognisers

- A Finite State Recogniser is a simple computing machine that reads (or tries to read) a sequence of symbols from an *input tape*.
- An FSA recognises (or accepts) a string of symbols (or words) S1, S2,…,Sn if starting in an initial state, it can read the symbols one after the other while making transitions from one state to another, until the transition reading in the last symbol takes the machine into a final state.

Final State Machines- Recognisers

An FSA, therefore, fails to recognise a string if

- It cannot reach a final state

Or

- It can reach a final state but there are still unread symbols left over when it does.

A Finite-State Based System

- The user is taken through a dialogue through a sequence of predetermined states:

-State 1: Execute Action a1. Ask for information via prompt p1.

The User phrase is recognised as response r1, which leads to State 2.

-State 2: Execute action a2. Ask for information via prompt p2.

Until

The User phrase is recognised as response rn, in which case execute action an+1 and ask for information via prompt pn+1.

A Finite State dialogue model example

- A Simple Travel inquiry system

- State 1: Destination?→State 2: Origin?→State 3Date?→State 4:Time?

When state 1 is matched, the system can move to State 2.

A Finite State dialogue model example

System: What is your destination?

- State 1: Destination?→State 2: Origin?→State 3: Date?→State 4:Time?

When state 1 is matched, the system can move to State 2.

A Finite State dialogue model example

System: What is your destination?

User: Paris

- State 1: Destination? Destination matched→ State 2: Origin?→State 3Date?→State 4:Time?

When state 1 is matched, the system can move to State 2.

A Finite State dialogue model example

System: What is your destination?

User: Paris

System: What is your origin?

- State 1: Destination? Destination matched→ State 2: Origin?→State 3: Date?→State 4:Time?

When state 1 is matched, the system can move to State 2.

A Finite State dialogue model example

System: What is your destination?

User: Paris

System: What is your origin?

User: London

- State 1: Destination? Destination matched→ State 2: Origin?→State 3: Date?→State 4:Time?

When state 1 is matched, the system can move to State 2.

A Finite State dialogue model example

The dialogue will be completed like this:
System: What is your destination?
User: Paris
System: What is your origin?
User: London
System: What day do you want to travel?
User: Saturday
System: What is your departure time?
User: 9.00 am

- State 1: Destination? Destination matched→ State 2: Origin?→State 3: Date?→State 4:Time?

When state 1 is matched, the system can move to State 2.

Advantages of FSA applications

- User’s input is limited to single predefined words or phrases.

-The approach can be used for either text-based or simple speech recognition

-Full natural language processing is not necessary

- Structured dialogue

- Simplified dialogue management component (user is directed through dialogue)
- More reliable performance (quite robust!)
- Simple to develop and suitable for well-structured tasks

Disadvantages

- User's input is limited to single pre-defined words or phrases

-Unable to cope with more complex dialogues, in particular:

Deviations from dialogue structure

Error recovery and dialogue repair

Dealing with non-atomic structures

Verification of speech recognition

- User cannot take initiative

-Grounding behaviour must be hard-coded (verification)

-Negotiation cannot be modelled (constraints may be unknown by system and user at outset).

- Thank you!