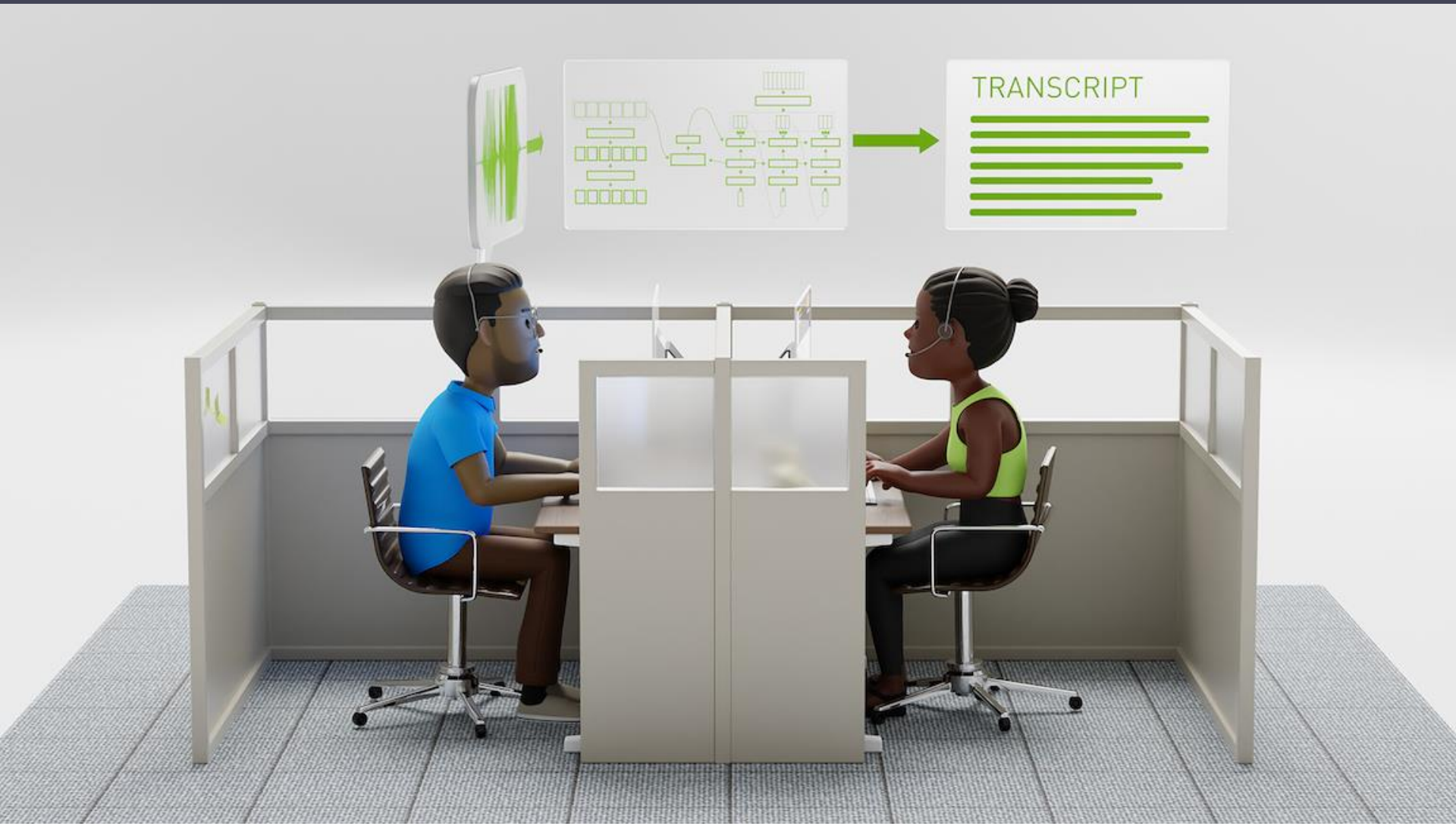
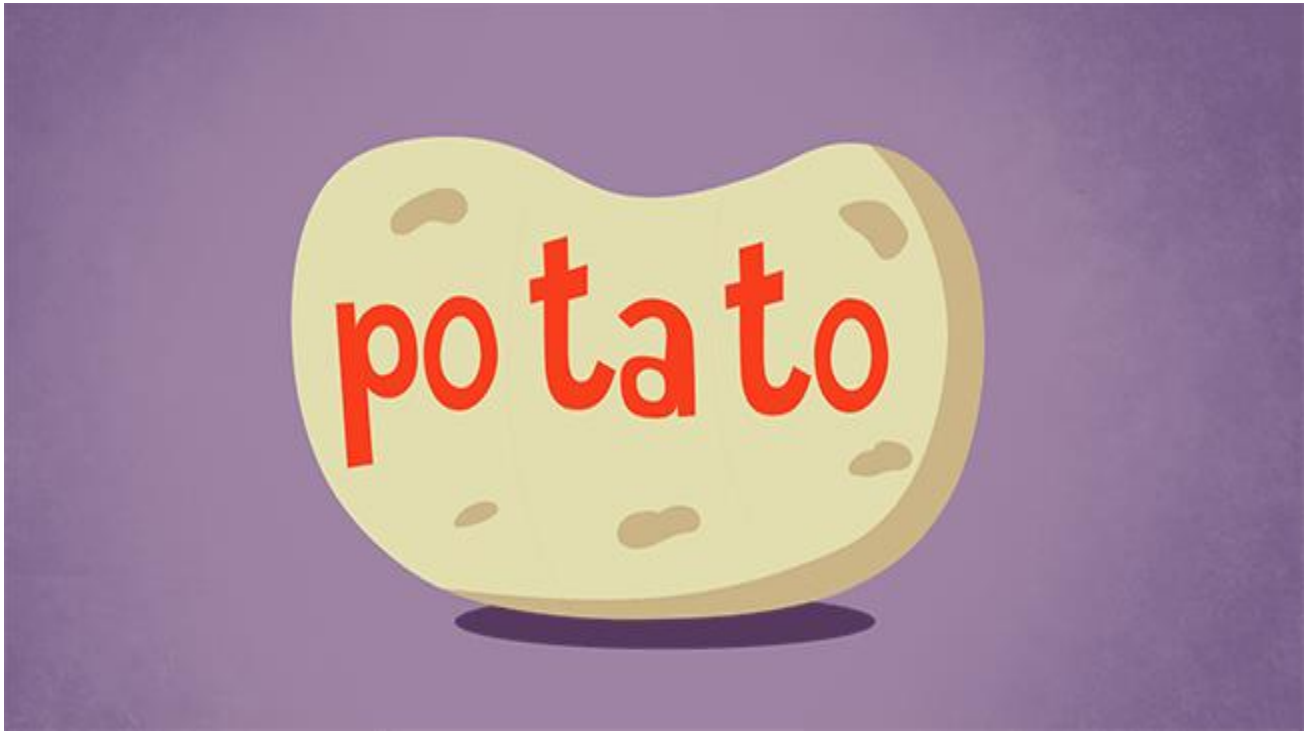
Natural Language Processing



Hazina A J
Asst prof.UOK

Phonology?



Phonology Definition

- Study of the **sound system** of a language.
- A language's sound system is made up of a set of phonemes which are used according to phonological rules.
- Describes sound contrasts which create differences in meaning within a language.

- Contd.
- For example, the phoneme /ɛ/ is different from the phoneme /i:/, so if we use the word *set* [sɛt] instead of *seat* [si:t], the **meaning** of the word will change.
- *slash* marks are used to indicate a phoneme /t/ (an abstract segment i.e. the representation of the sound).
- [t], used to indicate a phone (a physical segment i.e. the actual sound produced).

- Cont..
- The word *potato*: - In British English it is pronounced **po-tayh-to** [pəˈteɪtəʊ].- In American English it is pronounced **po-tay-to** [pəˈteɪˌtoʊ].

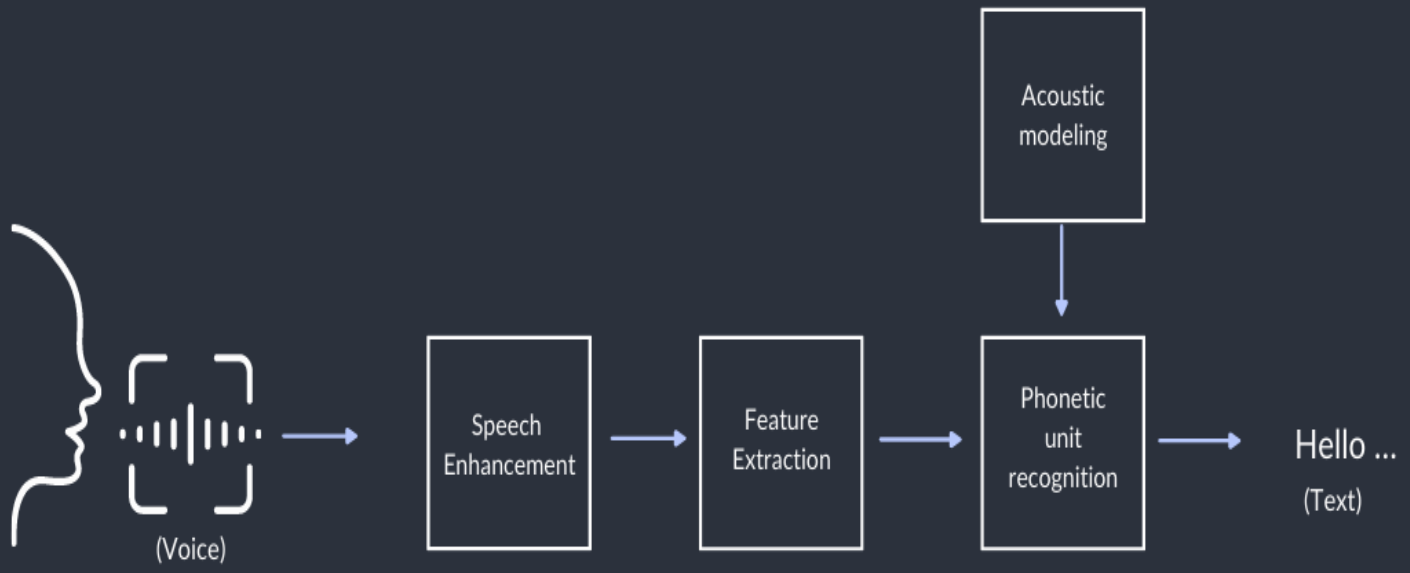
AUTOMATIC SPEECH RECOGNITION(ASR)

- Speech recognition focuses on the translation of speech from a verbal format to a text one whereas voice recognition just seeks to identify an individual user's voice.

AUTOMATIC SPEECH RECOGNITION(ASR)

- Speech to text conversion: AI converts the original audio file of the customer's speech into text.
- Natural Language Understanding (NLU): AI analyzes and processes text to create actionable instructions.
- Content relevance: AI comes back the best information that can help customers.

Speech Recognition



KARDOME

- The first step involves the computer identifying phonemes. These are the slightest sounds humans can make with their voices. The machine runs your voice through a stenograph as you speak into a microphone. This tool recognizes the phonemes in your voice.

- It then uses Natural Language Processing (NLP) to translate phonemes into readable text. It does that by comparing those recordings against databases of stored transcriptions.

Components of an ASR System

- Feature Extraction
- Acoustic Modeling
- Language Model
- Classification/Scoring

Feature Extraction

- Feature extraction extracts features from audio recordings.
- Think of features as word fingerprints that help identify spoken words.
- They identify specific characteristics such as pitch, volume, and accent.

Acoustic Modeling

- This model turns extracted features into a statistical parametric speech model.
- It will then compare against other models based on likelihood ratios.

Language Model

- A language model helps the machine determine which word sequences are possible.
- It uses grammar rules and probabilities for certain sounds occurring together within sentences.

Classification and Scoring

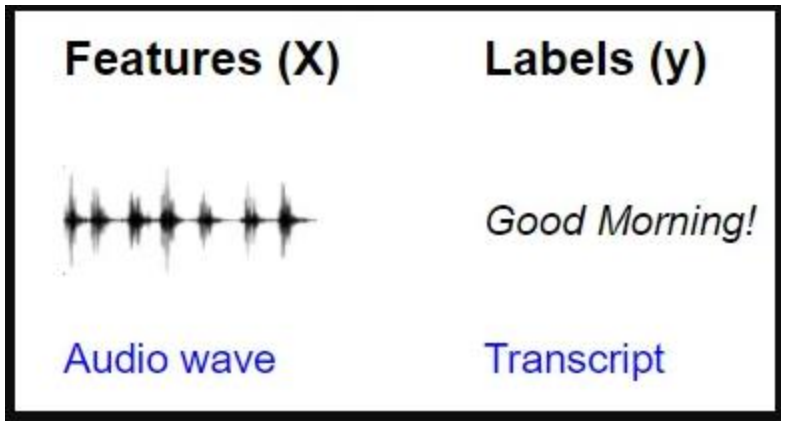
- A fancy term that means we've taken everything above and determined whether or not it was correct.
- If not, try again until you get it right. Once a system has gotten to that point, it will read through your data. It will extract features and make comparisons between models. Then it will decide on a final result.
- This process will repeat many times per second. Once it's reached an acceptable level of accuracy, it will move on.

APPLICATIONS

- Virtual assistants and chatbots
- Voice search
- Text-to-speech engine
- In-vehicle command
- Speech to text
- Enhanced security through voice recognition

How?

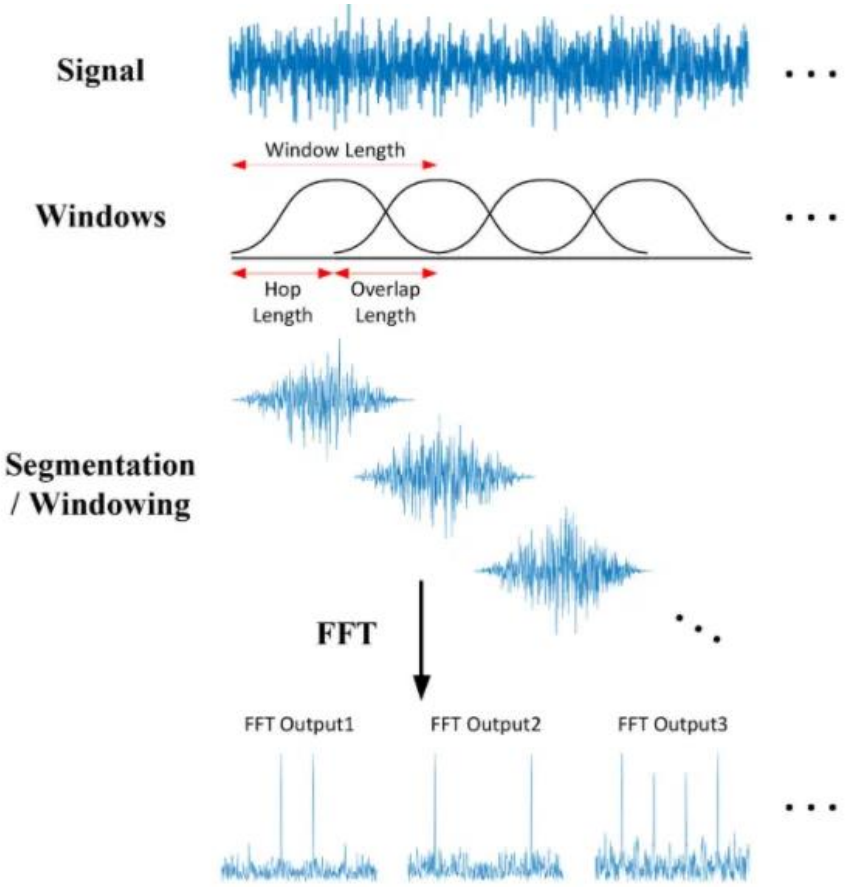
- Audio classification start with a sound clip and predict which class that sound belongs to, from a given set of classes. For Speech-to-Text problems, your training data consists of:
- Input features (X): audio clips of spoken words
- Target labels (y): a text transcript of what was spoken



Contd..

- key focus is finding the most probable word sequence given the audio.
- In other words, the principle is simplified to finding the word sequence *W* with the highest probability given the observed audio signals.

The division using fft



Word sequence: $W = w_1, w_2, \ldots, w_m$

Acoustic observations: $X = x_1, x_2, \ldots, x_n$

$$W^* = \arg\max_W P(W \mid X)$$ discriminative model

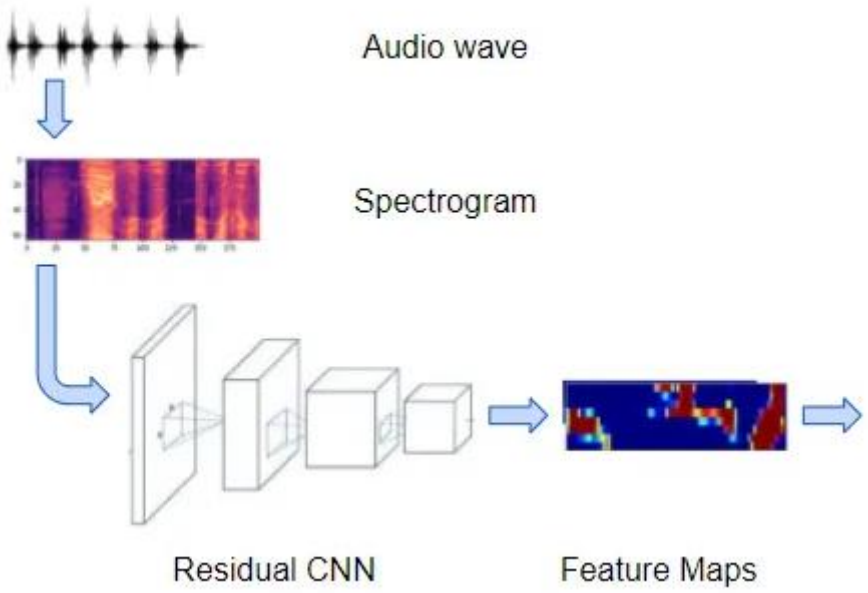
$$= \arg\max_W P(X \mid W)\, P(W) \,/\, P(X)$$

$$= \arg\max_W \underbrace{P(X \mid W)}_{\text{acoustic model}}\, \underbrace{P(W)}_{\text{language model}}$$ generative model

cont..

- The goal of the model is to learn how to take the input audio and predict the text content of the words and sentences that were uttered.
- Architectures mainly used based on deep learning.
- A CNN (Convolutional Neural Network) plus RNN-based (Recurrent Neural Network) architecture.

Simple architecture for CNN



Hidden Markov Models (HMM)

- An HMM is a statistical model.
- It maps the probability of random variables into a known variable.
- These observations can be acoustic signals representing words spoken by a person.

- Markov processes are commonly used to model sequential data, like text and speech.
- The **Hidden Markov Model (HMM)** is an extension of the Markov process used to model phenomena where the **states are hidden** or latent, but they **emit observations.**

HMM IN STT(speech-to-text)

- In a speech recognition system ,the states represent the actual text words to predict, but they are not directly observable.
- Hidden Markov Model has emission probabilities, which represent the probability that a particular state emits a given observation.

- Simple example

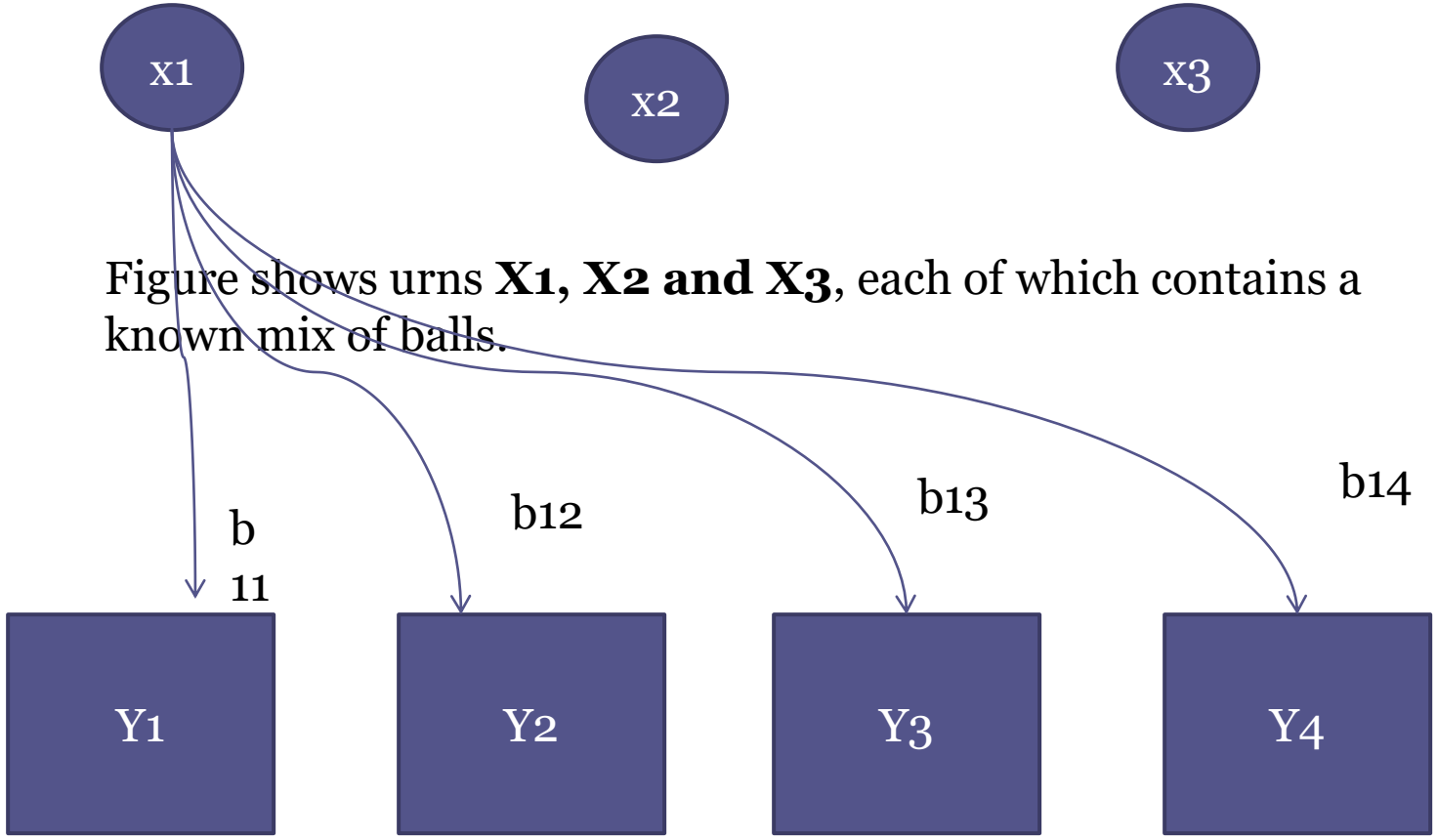
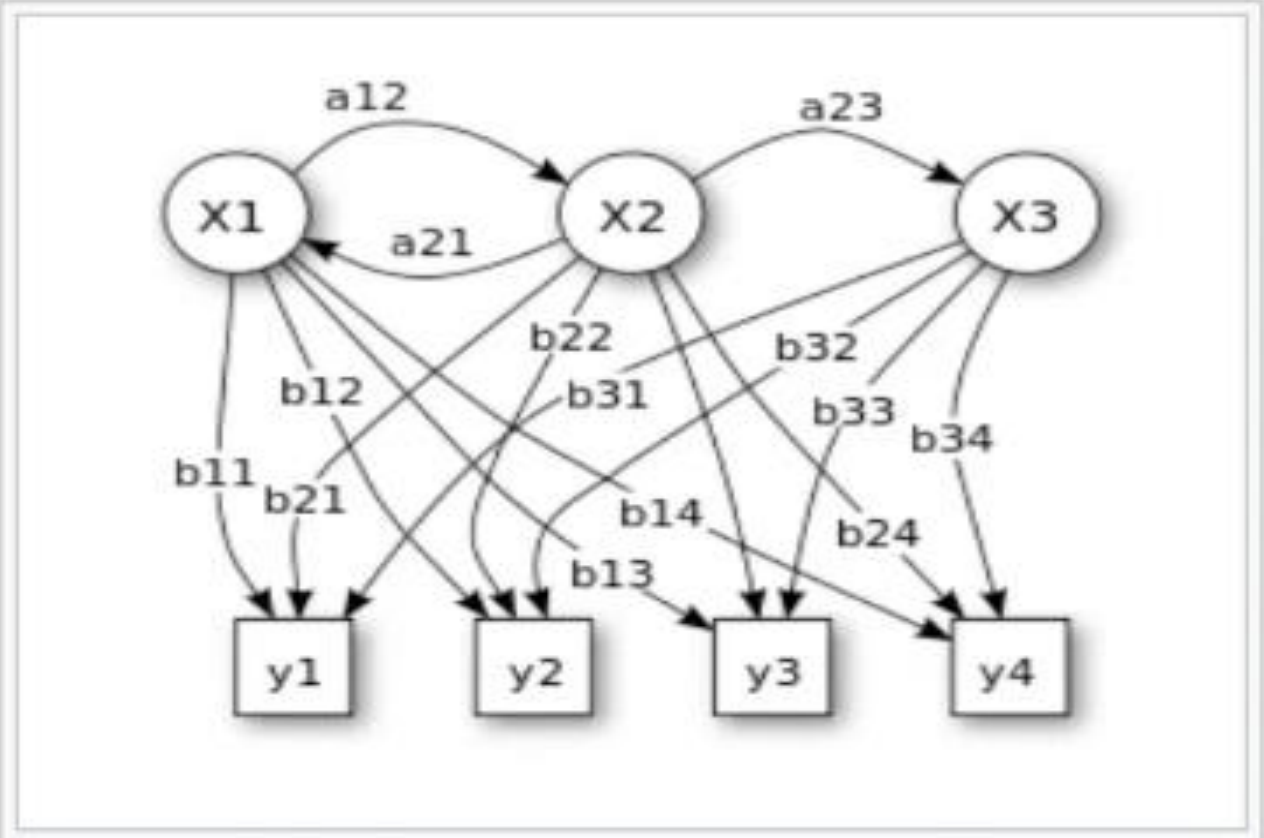


Figure shows urns **X1, X2 and X3**, each of which contains a known mix of balls.

each ball labeled **y1, y2, y3 and y4**

- A sequence of four balls is randomly drawn.
- i.e, b11,b12,b13,b14…b34.
- **The user observes a sequence of balls y1,y2,y3 and y4 and is attempting to discern the hidden state which is the right sequence of three urns that these four balls were pulled from.**



X – states

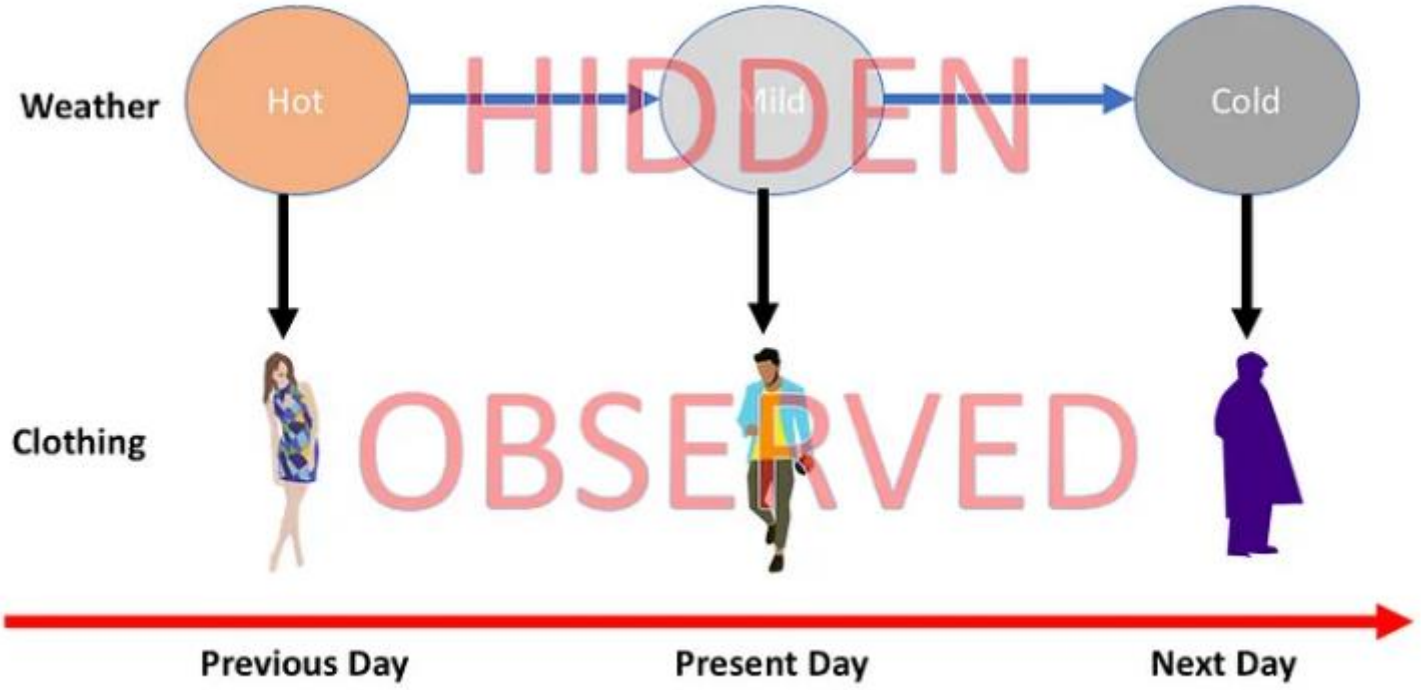
y – possible observations

a – state transition probabilities

b – output probabilities

- Consider the scenario below where the weather, the hidden variable, can be hot, mild or cold and the observed variables are the type of clothing worn.
- The arrows represent transitions from a hidden state to another hidden state or from a hidden state to an observed variable.
- Note: Markov assumption, each state only depends on the previous state and not on any other prior states

Weather problem



- Once we know the joint probability of a sequence of hidden states, we determine the best possible sequence i.e. the sequence with the highest probability and choose that sequence as the best sequence of hidden states.

In order to compute the joint probability of a sequence of hidden states, we need to assemble three types of information.

- Generally, the term "states" are used to refer to the hidden states and "observations" are used to refer to the observed states.
- **Transition data** — the probability of transitioning to a new state conditioned on a present state.
- **Emission data** — the probability of transitioning to an observed state conditioned on a hidden state.
- **Initial state information** — the initial probability of transitioning to a hidden state. This can also be looked at as the prior probability.