# Multimodal AI

Semester-VII

August 17, 2024

# Eval Scheme- Theory

| Component | Particular | Marks/Weightage |
|---|---|---|
| 1 | Quiz | 10 |
| 2 | UT | 10 |
| 3 | Seminar/Assignment | 10 |

# Module-I
# Fundamentals of Multimodal AI, Challenges, and Applications

# What is Multimodal AI/System?

- Multimodal AI, or multimodal systems, refers to artificial intelligence systems designed to process and understand information from multiple modalities or types of data simultaneously.
- These modalities can include text, images, audio, video, and more.
- The goal of multimodal AI is to create systems that can interpret and generate information across these different forms of data, enabling richer and more comprehensive understanding and interaction with the world.

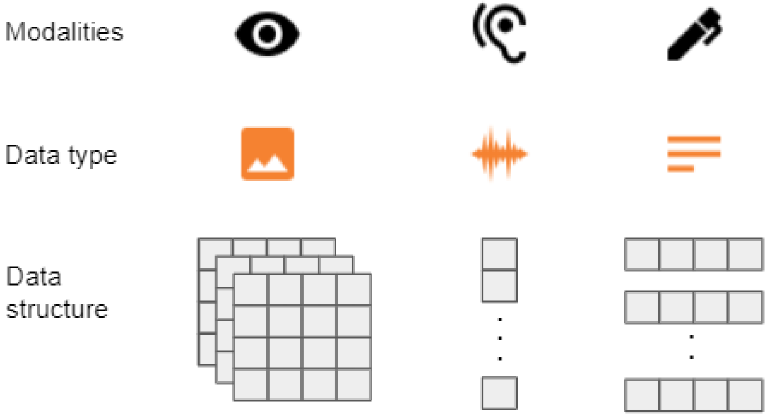
# Historical View

- Early systems focused on single modalities (e.g., text or image processing).
- Advances in computing power and machine learning enabled the integration of multiple modalities.
- Significant progress in the 2000s with the advent of deep learning.

# Multimodal vs Multimedia

- **Multimedia**: Content that combines different content forms such as text, audio, images, animations, video, and interactive content.
- **Multimodal**: Systems that can process and integrate multiple forms of information from different modalities.
- Basically, Multimedia+AI = Multimodal AI

# Modalities & Data Structures

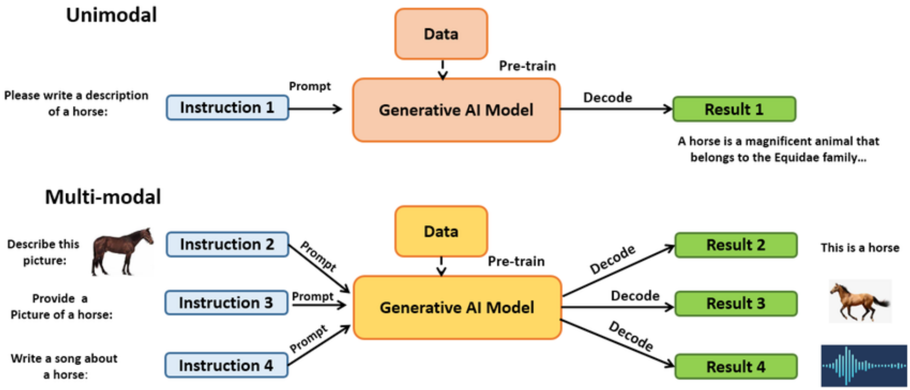


Modalities

Data type

Data structure

# Unimodal vs Multimodal



# Why Multimodal?

- Enhances the understanding and context of information.
- Mimics human cognitive abilities that rely on multiple senses.
- Improves performance in tasks like language understanding, scene recognition, and human-computer interaction.

# Enhanced Understanding and Context

- Multimodal AI can provide a richer understanding by combining different types of data.
- For example, combining text and images can improve comprehension in applications like image captioning.
- Audio-visual integration can enhance speech recognition systems, especially in noisy environments.

# Improved Accuracy and Robustness

- Integrating multiple data sources may lead to more accurate and reliable predictions.
- For instance, in medical diagnosis, combining MRI scans (visual) with patient history (textual) can improve diagnostic accuracy.
- Multimodal systems are less likely to fail when one modality is compromised.

# Enhanced User Interaction and Experience

- Multimodal AI enables more natural and intuitive user interactions.
- Voice assistants that combine speech recognition with visual feedback (e.g., smart displays) provide a richer user experience.
- Augmented reality (AR) applications benefit from combining real-world visuals with contextual information.

# Applications in Real-World Scenarios

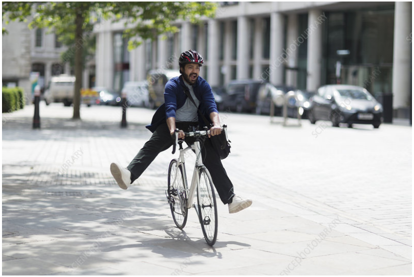
- Healthcare: Multimodal AI can integrate patient data from various sources for comprehensive analysis.
- Autonomous vehicles: Combining visual data (cameras) with spatial data (LIDAR) for better navigation.
- Customer service: Chatbots that use both text and voice to interact with users.

# Image Captioning

- Generating textual descriptions for images.
- Utilizes both visual and textual data.
- Applications: Accessibility, search engines, social media.

# Image Captioning: Example

- **Example Image**



- **Generated Captions**
  - *”A group of people standing around a table with food.”*
  - *”A child playing with a dog in the park.”*
  - *”A man riding a bike on a city street.”*
- **Model Architecture**
  - Uses Convolutional Neural Networks (CNNs) to extract features from the image.
  - Uses Recurrent Neural Networks (RNNs) or Transformers to generate descriptive text based on the extracted features.

# Image Captioning: Detailed Workflow

- **Feature Extraction**
  - Input image is passed through a pre-trained CNN (e.g., ResNet, Inception) to extract high-level features.
  - These features represent the essential visual aspects of the image.
- **Caption Generation**
  - Extracted features are fed into an RNN or Transformer model.
  - The model generates a sequence of words to form a caption, predicting one word at a time.
  - Attention mechanisms may be used to focus on different parts of the image during the caption generation process.
- **Training**
  - Requires a large dataset of images with corresponding captions (e.g., MS COCO).
  - Model learns to map visual features to textual descriptions through supervised learning.

# Popular Multimodal models for Image Captioning

- **Vision Transformers (ViTs)**: Combining the power of transformers with image processing.
- **Multimodal Transformers**: Models like CLIP (Contrastive Language–Image Pretraining) by OpenAI that align visual and textual embeddings.
- **Generative Models**: Models like GPT-4, which can handle multimodal inputs and generate descriptive text.

# FLAVA for Image Captioning[1]

- FLAVA (Foundational Language and Vision Alignment) is a model that handles multimodal tasks, including image captioning.
- Developed by Facebook AI, FLAVA aims to create a unified framework that can understand and generate visual and textual information, integrating the capabilities of image processing and natural language understanding.

[1] `https://arxiv.org/abs/2112.04482`

# Key features of FLAVA

- **Unified Multimodal Architecture:**
  FLAVA employs a single architecture that can handle visual and textual inputs, making it versatile for various tasks beyond image captioning, such as visual question answering and multimodal classification.
- **Cross-Modal Attention:**
  It uses cross-modal attention mechanisms to effectively combine information from images and text. This allows the model to generate coherent and contextually relevant captions based on the visual input.

# Challenges in Image Captioning

- **Complexity of Visual Scenes**
  - Capturing and describing detailed and complex scenes accurately.
- **Diversity in Language**
  - Generating natural and diverse captions that accurately reflect the content of the image.
- **Context Understanding**
  - Understanding the context and relationships between objects in the image.
- **Data Annotation**
  - Requires large, annotated datasets for training, which can be time-consuming and expensive to create.

# Video Description

- Creating textual narratives for video content.
- Combines visual data, motion, and often audio.
- Applications: Media, entertainment, education.

# Audio-Visual Speech Recognition (AVSR)

- Recognizing speech using both audio and visual cues (e.g., lip movements).
- Improves accuracy in noisy environments.
- Applications: Assistive technologies, transcription services.

# AVSR: Example

- **Example Scenario**
  - In a noisy environment, traditional audio-only speech recognition might struggle to transcribe speech accurately.
  - By incorporating visual information of the speaker's lip movements, AVSR systems can improve the transcription accuracy.
- **Sample Output**
  - *"The quick brown fox jumps over the lazy dog."* (Correct transcription with AVSR)
  - *"The quick brown f*x jumps over the l*zy dog."* (Possible transcription with audio-only)

# AVSR: Detailed Workflow

- **Audio Feature Extraction**
  - Extracts features such as MFCCs (Mel-Frequency Cepstral Coefficients) from the audio signal.
- **Visual Feature Extraction**
  - Uses Convolutional Neural Networks (CNNs) to extract features from video frames focusing on lip movements.
- **Fusion of Audio and Visual Features**
  - Combines audio and visual features using techniques like concatenation, attention mechanisms, or multimodal transformers.

# Contd...

**Sequence Modeling**

- Uses models like Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRUs), or Transformers to model the temporal dynamics of the fused features.

**Decoding**

- Decodes the sequence model's output into text, often using beam search or other decoding strategies.

**Training**

- Requires synchronized audio-visual datasets for training, such as GRID or LRS3.
- Trains the system to map multimodal features to textual transcriptions through supervised learning.

# Downstream Task: Speech Recognition

- Recurrent Neural Networks (RNNs): Often used for sequence modeling, such as Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU) networks.
- Attention Mechanisms: To focus on important parts of the input sequence.
- Transformer Models: Recently, transformer-based models like BERT and GPT have shown great success in speech and language tasks.

# Training a speech recognition model

- Connectionist Temporal Classification (CTC loss) is designed for sequence-to-sequence problems where the alignment between input and output sequences is unknown.
- It is particularly useful for speech recognition, as it allows the model to output variable-length sequences.

# CTC Loss Function

Given:

- $x = (x_1, x_2, \ldots, x_T)$: input sequence (e.g., audio features).
- $y = (y_1, y_2, \ldots, y_U)$: target sequence (e.g., transcription).
- $\mathcal{B}(z)$: mapping function that removes blanks and repeated characters from $z$.

# Softmax Output

The network outputs a probability distribution over labels (including a special blank token) for each time step:

$$\mathrm{p}(t) = \mathrm{softmax}(\mathrm{Wh}_t + \mathrm{b})$$

where $\mathrm{h}_t$ is the hidden state at time $t$, and W and b are the weight matrix and bias vector.

# CTC Path Probability

Define a path $\pi$ as a possible alignment of the target sequence with the input sequence, including blank tokens. The probability of a path $\pi$ is the product of the probabilities of each label at each time step:

$$P(\pi|x) = \prod_{t=1}^{T} p(\pi_t|x_t)$$

# Summing Over All Paths

The probability of the target sequence $y$ is the sum of the probabilities of all valid paths that map to $y$ using the mapping function $\mathcal{B}$:

$$P(y|x) = \sum_{\pi \in \mathcal{B}^{-1}(y)} P(\pi|x)$$

# CTC Loss

The CTC loss is the negative log probability of the target sequence given the input sequence:

$$\mathcal{L}_{\mathrm{CTC}} = -\log P(y|x)$$

# Summary for Multimodal Speech Recognition

- CTC loss enables training of RNNs for tasks with unaligned input and output sequences.
- It is particularly useful for speech recognition and other sequence-to-sequence problems.

# Challenges in AVSR

- **Synchronization**
    - Ensuring precise alignment between audio and visual data streams.
- **Variability in Visual Data**
    - Handling differences in lighting, occlusions, and speaker variations.
- **Noisy Environments**
    - Effectively filtering out background noise while capturing essential speech signals.
- **Computational Complexity**
    - Managing the increased computational load due to processing multiple data streams.
- **Data Requirements**
    - Requires large, annotated, synchronized audio-visual datasets for effective training.

# The McGurk Effect

- The McGurk effect denotes a phenomenon of speech perception.
- A listener attends to mismatched audio and visual stimuli and perceives an illusory third sound, typically a conflation of the audio-visual stimulus.
- This multimodal interaction has been exploited in various English-language experiments.

# Representation Learning

- Learning effective representations from multiple modalities.
- Ensuring that representations capture the relevant information.
- Techniques: Autoencoders, Transformers, Convolutional Neural Networks.

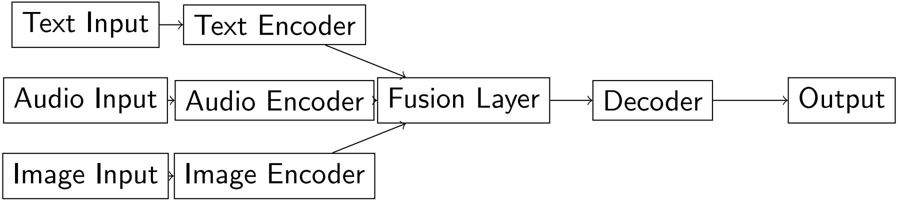
# Importance of Representation Learning

- **Feature Extraction**:
  - Automatically learns features from raw data.
  - Reduces the need for manual feature engineering.
- **Improved Performance**:
  - Enhances the performance of downstream tasks like classification, regression, and clustering.
- **Generalization**:
  - Learns representations that generalize well to new, unseen data.
- **Multimodal Integration**:
  - Facilitates the integration of diverse types of data, enhancing overall system understanding.

# Techniques in Multimodal Representation Learning

- **Autoencoders**
  - Use neural networks to learn efficient codings of input data.
  - Can be applied to individual modalities and then combined for multimodal data.
- **Transformers**
  - Utilize self-attention mechanisms to learn contextual representations.
  - Can handle multiple modalities by processing and integrating them within a unified architecture.
- **Multimodal Embeddings**
  - Learn shared latent spaces where information from different modalities is projected.
  - Examples include joint text-image embeddings (e.g., CLIP) and audio-visual embeddings.

# Multimodal Transformer Architecture



- Each modality has its own encoder to process the input data.
- The Fusion Layer combines features from all modalities.
- The Decoder generates the final output from the fused representation.

# Fusion Strategies

- **Early Fusion**
  - Combines raw data from different modalities before feature extraction.
  - Example: Concatenating pixel values from images and raw audio signals.
- **Late Fusion**
  - Combines high-level features extracted from each modality separately.
  - Example: Merging the outputs of CNNs (for images) and RNNs (for text) at a later stage.
- **Hybrid Fusion**
  - Integrates both early and late fusion strategies.
  - Example: Combining low-level and high-level features from multiple modalities.

# Applications of Multimodal Representation Learning

- **Image Captioning**
  - Learns representations that connect visual data with textual descriptions.
- **Video Analysis**
  - Integrates visual and auditory information to understand and annotate video content.
- **Sentiment Analysis**
  - Combines textual data with facial expressions and vocal tones to assess sentiment.
- **Healthcare**
  - Merges medical imaging, patient records, and genomic data for comprehensive diagnostics.

# Challenges in Multimodal Representation Learning

- **Heterogeneity of Data**
    - Different modalities have diverse characteristics and distributions.
- **Alignment**
    - Synchronizing data from multiple modalities in time and space.
- **Scalability**
    - Managing the computational complexity of processing large-scale multimodal data.
- **Interpretability**
    - Understanding and explaining how multimodal representations are learned and used.

# Translation

- Converting information from one modality to another (e.g., speech-to-text).
- Challenges include maintaining context and meaning.
- Techniques: Sequence-to-sequence models, neural machine translation.

# What is Neural Machine Translation?

- Neural Machine Translation (NMT) is a type of machine translation that uses neural networks to predict the likelihood of a sequence of words.
- It overcomes many limitations of traditional statistical machine translation.
- NMT models are typically end-to-end, meaning they learn to translate directly from source to target text.

# Components of NMT

- Encoder-Decoder Architecture
- Sequence-to-Sequence (Seq2Seq) Model
- Attention Mechanism

# Role of Multimodal AI in NMT

- Incorporates visual and auditory information to improve translation accuracy.
- Examples include image captions, video subtitles, and speech translation.

# Applications of Multimodal NMT

- Image captioning and translation.
- Video subtitling in multiple languages.
- Speech-to-text translation with context from visual inputs.

# Alignment

- Synchronizing information from different modalities.
- Ensuring temporal and contextual coherence.
- Techniques: Dynamic Time Warping, Attention Mechanisms.

# What is Alignment?

- Alignment in multimodal AI refers to the process of synchronizing information from different modalities such as text, images, and audio.
- It ensures that the temporal and contextual coherence is maintained across these different types of data.

# Importance of Alignment

- Proper alignment enhances the performance of multimodal AI systems by providing a unified understanding of the data.
- It is crucial for applications like image captioning, video subtitling, and speech-to-text translation, where multiple modalities are involved.

# How can we quantify alignment in Multimodal AI use-case?

1. Attention mechanisms can be used to compute alignment scores between text and image embeddings.
2. You can evaluate alignment using cross-modal retrieval tasks. This involves using one modality (e.g., text) to retrieve related instances from another modality (e.g., images).
3. Contrastive learning techniques can be used to align embeddings from different modalities by bringing related embeddings closer and pushing unrelated embeddings apart.
4. CCA is a statistical method that can be used to find linear relationships between two sets of variables (e.g., text and image embeddings).

# Synchronizing Information from Different Modalities

- Combining data from text, images, and audio to create a comprehensive representation.
- Ensuring that the information from one modality complements and enhances the understanding of information from another modality.

# Ensuring Temporal and Contextual Coherence

- Temporal coherence ensures that events occurring simultaneously in different modalities are aligned correctly in time.
- Contextual coherence ensures that the information presented across modalities is contextually consistent and meaningful.

# Introduction to Cosine Similarity

Cosine similarity is a measure of similarity between two non-zero vectors in an inner product space.

- Measures the cosine of the angle between two vectors.
- Ranges from -1 to 1.
- Used in various applications, including text and image alignment.

# Cosine Similarity Formula

For two vectors a and b, cosine similarity is given by:

$$\mathrm{sim}(\mathrm{a}, \mathrm{b}) = \frac{\mathrm{a} \cdot \mathrm{b}}{\|\mathrm{a}\| \|\mathrm{b}\|}$$

where:

- $\mathrm{a} \cdot \mathrm{b}$ is the dot product of a and b.
- $\|\mathrm{a}\|$ and $\|\mathrm{b}\|$ are the magnitudes (norms) of a and b respectively.

# Example Calculation

Consider vectors $a = (1, 2, 3)$ and $b = (4, 5, 6)$.

- Dot Product:

$$a \cdot b = 1 \cdot 4 + 2 \cdot 5 + 3 \cdot 6 = 32$$

- Magnitudes:

$$\|a\| = \sqrt{1^2 + 2^2 + 3^2} = \sqrt{14}$$

$$\|b\| = \sqrt{4^2 + 5^2 + 6^2} = \sqrt{77}$$

- Cosine Similarity:

$$\text{sim}(a, b) = \frac{32}{\sqrt{14} \cdot \sqrt{77}} \approx 0.974$$

# Application to Multimodal Alignment

In multimodal AI, cosine similarity can quantify alignment between embeddings from different modalities:

- **Text Embeddings**: Represent semantic information from text.
- **Image Embeddings**: Capture visual features from images.
- **Goal**: Measure how well text and image embeddings align.

# Practical Example

Suppose we have embeddings from a text model and an image model.

- Text embedding: t
- Image embedding: v
- Cosine Similarity:

$$\mathrm{sim}(\mathrm{t}, \mathrm{v}) = \frac{\mathrm{t} \cdot \mathrm{v}}{\|\mathrm{t}\|\|\mathrm{v}\|}$$

This value helps in understanding how closely related the text and image are, which is useful in tasks like image captioning and cross-modal retrieval.

# What is Dynamic Time Warping?

- DTW is a technique used to measure the similarity between two temporal sequences which may vary in speed.
- It finds an optimal alignment between the sequences by stretching or compressing the time dimensions.

$$DTW(i,j) = \min\{DTW(i-1,j), DTW(i,j-1), DTW(i-1,j-1)\} + d(i,j) \tag{1}$$

- Where $d(i,j)$ is the distance between points $i$ and $j$.

# Introduction to DTW

Dynamic Time Warping (DTW) is an algorithm for measuring similarity between two temporal sequences.

- Handles sequences that may vary in speed.
- Useful in time series analysis and pattern recognition.

# Distance Matrix

Let x $= (x_1, x_2, \ldots, x_N)$ and y $= (y_1, y_2, \ldots, y_M)$ be two sequences.

## Distance Function

The distance between elements is defined as:

$$d(i,j) = \|x_i - y_j\|$$

## Distance Matrix

The distance matrix $D$ is:

$$D(i,j) = d(i,j)$$

# Cumulative Cost Matrix

Define the cumulative cost matrix $C$, where $C(i,j)$ is the minimum cumulative distance to align $\mathrm{x}_{1:i}$ with $\mathrm{y}_{1:j}$.

$$C(i,j) = D(i,j) + \min \left( \begin{array}{l} C(i-1,j) \\ C(i,j-1) \\ C(i-1,j-1) \end{array} \right)$$

Base cases:

$$C(0,0) = D(0,0)$$

$$C(i,0) = D(i,0) + C(i-1,0)$$

$$C(0,j) = D(0,j) + C(0,j-1)$$

# DTW Distance

The DTW distance between sequences x and y is given by:

$$\mathrm{DTW}(\mathrm{x}, \mathrm{y}) = C(n-1, m-1)$$

# Application of DTW in Multimodal AI

- DTW is used to synchronize data from different modalities (e.g., audio and video) that may not be perfectly aligned in time.
- Example: Aligning audio speech with lip movements in a video.

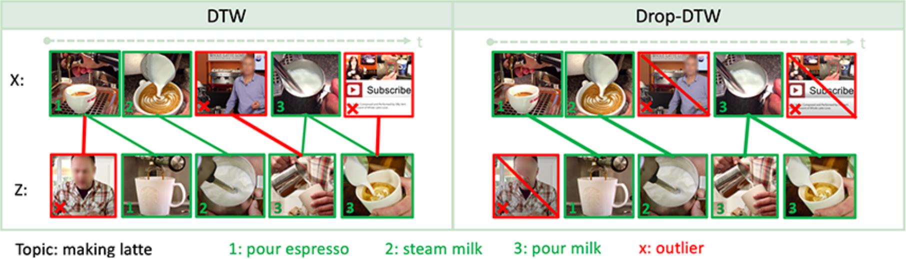


Figure: DTW aligning audio and video sequences

# DTW Example

Consider sequences x = $(1, 2, 3)$ and y = $(2, 2, 4)$. Compute the distance matrix $D$, cost matrix $C$, and DTW distance.

- Distance matrix $D$:

$$D = \begin{bmatrix} |1-2| & |1-2| & |1-4| \\ |2-2| & |2-2| & |2-4| \\ |3-2| & |3-2| & |3-4| \end{bmatrix}$$

- Cost matrix $C$:

$$C = \begin{bmatrix} C(1,1) & C(1,2) & C(1,3) \\ C(2,1) & C(2,2) & C(2,3) \\ C(3,1) & C(3,2) & C(3,3) \end{bmatrix}$$

# Limitations of DTW

1. Linearity: Traditional DTW can only handle linear alignments, while multimodal data often exhibits non-linear relationships.
2. High dimensionality: Multimodal data can have high dimensionality, which can make DTW computationally expensive and less effective.

# Example Code for DTW[2]

- To compute Dynamic Time Warping (DTW) between text and image embeddings, you'll first need to generate the embeddings for both the text and images.
- For text embeddings, models like BERT or Sentence Transformers can be used. For image embeddings, models like ResNet or VGG can be used.
- Once you have these embeddings, you can use a library like fastdtw/dtw to compute the DTW.

## Refer code

`https://colab.research.google.com/drive/1kVxj79oFyROEfZ6uQvxCzgkRxZME_h-t?usp=sharing`

[2] Read more at `https://medium.com/@markstent/dynamic-time-warping-a8c5027defb6`

# Fusion

- Combining information from multiple modalities into a single representation.
- Enhances the robustness and accuracy of the system.
- Techniques: Early fusion, late fusion, hybrid fusion.

# Co-learn

- Joint learning from multiple modalities.
- Leveraging shared and complementary information.
- Techniques: Multi-task learning, co-training.

# Joint learning

1. Joint learning in multimodal AI refers to training a single model to simultaneously process and learn from multiple modalities.
2. This approach aims to create a shared representation space where different modalities are aligned and can be used to solve various tasks.

# Conclusion

- Multimodal systems integrate various types of data to improve performance and user experience.
- Challenges remain in representation, translation, alignment, fusion, and co-learning.
- Continued advancements are crucial for more sophisticated and human-like AI systems.

# Scan QR for Quiz-based attendance