ECE 417 Lecture 1: Multimedia Signal Processing

Mark Hasegawa-Johnson

8/29/2017

Today’s Lecture

- Syllabus
- Overview of what we’ll learn this semester
- Review of important linear algebra concepts, part one
- Sample problem

Syllabus

- Go read the syllabus: http://courses.engr.Illinois.edu/ece417/

Overview of what we'll learn this semester

- What is multimedia?
- What is processing?
- The basic architecture: encoder, embedder, aligner, decoder
- Parametric learning: categories of methods, types of parameters that you can learn from data

What is multimedia?

- Audio, Video, and Images
- Defining feature #1: very big
  - Speech audio: 16000 samples/second
  - Multimedia audio: 44100 samples/second (often downsampled to 22050!)
  - Image: 6M pixels ~ 2000x3000 (often downsampled to 224x224)
  - Video: 640x480 pixels/image, 30 frames/second = 9,216,000 real numbers per second
- Why “very big” matters: you can’t train a neural net to observe a whole image
  - Rule of 5: there must be 5 training examples per trainable parameter
  - Two-layer neural net with 1024 hidden nodes, 6M inputs has 1024*6000000 trainable parameters = 6 billion trainable parameters
  - With 6 billion trainable parameters, you need 30 billion training examples
  - Result: we need some trick to reduce the number of parameters
  - This semester is all about finding tricks that work (much better)

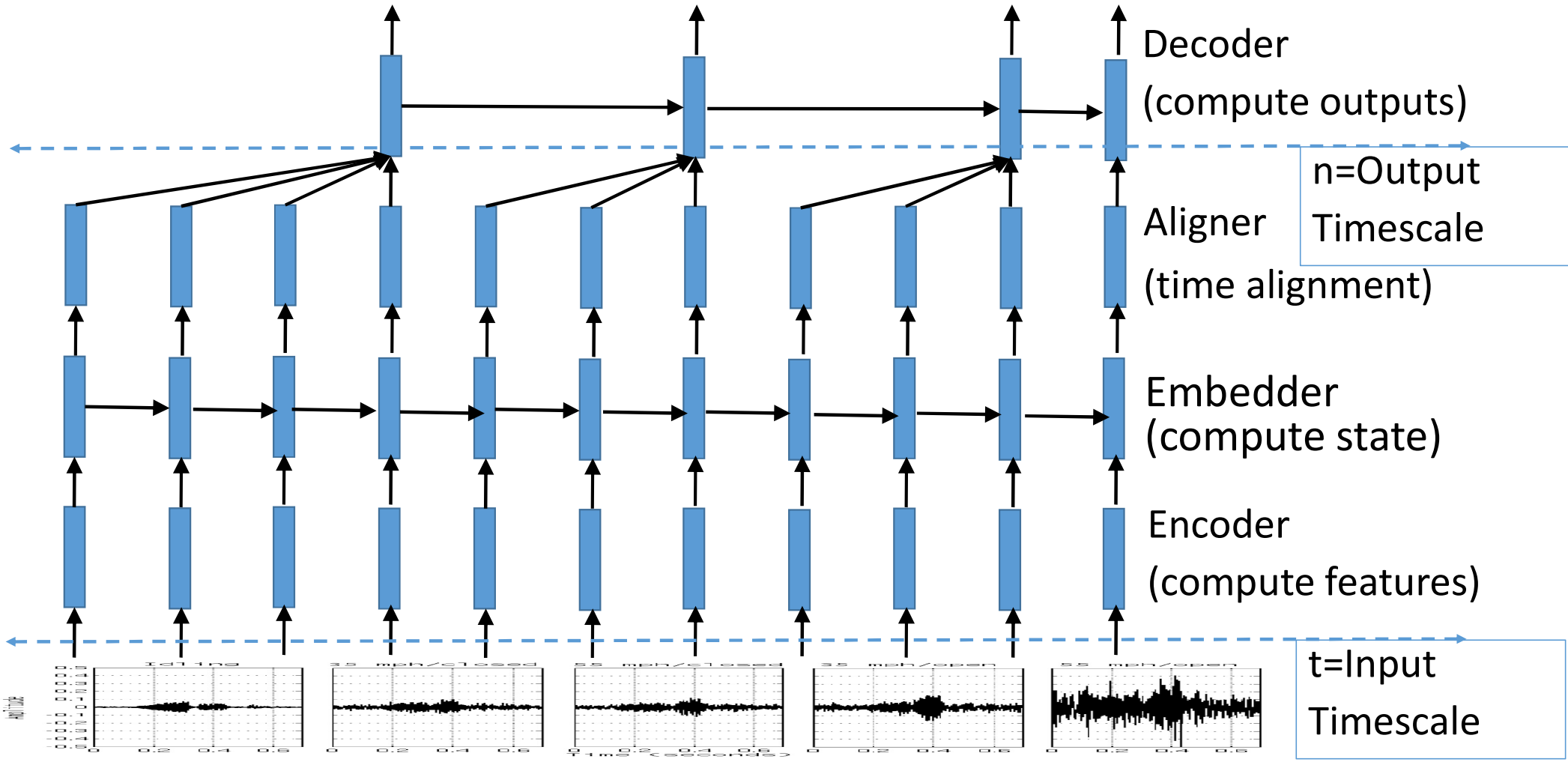
What is Multimedia: Defining Feature #2

- Defining feature #2: variable input dimension
  - Image classification, e.g., imagenet: we like to downsample every image to 224x224 first, then train a NN with 224*224=50176 inputs
  - Audio speech recognition: no can do
    - Why?
- Solution: streaming methods
  - “Recognition” instead of “Classification” --- variable number of inputs, variable number of outputs

Processing = convert from one type of signal to another

	Image	Video	Audio	Text (variable-length sequence of symbols)	Metadata (tags drawn from predefined set)
Image	X	Motion prediction Animation	Spoken captioning	Automatic captioning	Object recognition Face recognition
Video	Heat map Frame capture Summarization Shot boundary detection	X		Lipreading/ Speechreading Audiovisual speech recognition Summarization	
Audio	Spectrogram(STFT)	STFT Avatar animation	X	Automatic speech recognition	Music genre Emotion/sentiments Speaker ID
Text		Audiovisual speech synthesis	Speech synthesis	X	Sentiment
Metadata	Search	Search	Search		X

Basic Multimedia Architecture: EEAD



Parametric Learning

- What it means:
  1. Define some general family of functions, $\vec{y} = f(\vec{x}, W)$, where $\vec{x}$ is an input feature vector, and W is a set of learnable parameters. For example, in MP1, it will be something like $\vec{y} = [\frac{x_1}{w_1}, \frac{x_2}{w_2}, \dots]$
  2. "Learn" the parameters: choose W in order to maximize performance on some training dataset

Types of Parametric Learning

- Metric Learning/Feature Learning (MP1): Learn to convert input features into some new feature set that better matches human perception
- Classifier Learning (MP2, MP3): Learn to classify the input.
- Bayesian Learning (MP4, MP5): Learn to estimate how likely the input is, given some assumed class label.
- Deep learning (MP6, MP7): Combines ***feature learning*** and ***classifier learning***. Each layer computes features that are used by the next layer up.

Basics of Linear Algebra

- Vector Space
- Banach Space, Norm
- Hilbert Space, Inner Product
- Linear Transform
- Affine Transform

Vector Space

A vector space is a set, closed under addition, that satisfies:

- Addition commutativity
- Addition associativity
- Addition identity element
- Addition inverse
- Compatibility of scalar and field multiplication
- Multiplication identity element
- Distribution of multiplication over vector addition
- Distribution of multiplication over field addition

In this article, vectors are represented in boldface to distinguish them from scalars.[nb 1] In the two examples above, the field is the field of the real numbers and the set of the vectors consists of the planar arrows with fixed starting point and of pairs of real numbers, respectively.

To qualify as a vector space, the set V and the operations of addition and multiplication must adhere to a number of requirements called axioms.[1] In the list below let $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ be arbitrary vectors in V, and a and b scalars in F.

Axiom	Meaning
Associativity of addition	$\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$
Commutativity of addition	$\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$
Identity element of addition	There exists an element $\mathbf{0} \in V$, called the *zero vector*, such that $\mathbf{v} + \mathbf{0} = \mathbf{v}$ for all $\mathbf{v} \in V$.
Inverse elements of addition	For every $\mathbf{v} \in V$, there exists an element $-\mathbf{v} \in V$, called the *additive inverse* of $\mathbf{v}$, such that $\mathbf{v} + (-\mathbf{v}) = \mathbf{0}$.
Compatibility of scalar multiplication with field multiplication	$a(b\mathbf{v}) = (ab)\mathbf{v}$ [nb 2]
Identity element of scalar multiplication	$1\mathbf{v} = \mathbf{v}$, where 1 denotes the multiplicative identity in F.
Distributivity of scalar multiplication with respect to vector addition	$a(\mathbf{u} + \mathbf{v}) = a\mathbf{u} + a\mathbf{v}$
Distributivity of scalar multiplication with respect to field addition	$(a + b)\mathbf{v} = a\mathbf{v} + b\mathbf{v}$

These axioms generalize properties of the vectors introduced in the above examples. Indeed, the result of addition of two ordered pairs (as in the second example above) does not depend on the order of the summands:

$$(x_{\mathbf{v}}, y_{\mathbf{v}}) + (x_{\mathbf{w}}, y_{\mathbf{w}}) = (x_{\mathbf{w}}, y_{\mathbf{w}}) + (x_{\mathbf{v}}, y_{\mathbf{v}}).$$

Likewise, in the geometric example of vectors as arrows, $\mathbf{v} + \mathbf{w} = \mathbf{w} + \mathbf{v}$ since the parallelogram defining the sum of the vectors is independent of the order of the vectors. All other axioms can be checked in a similar manner in both examples. Thus, by disregarding the concrete nature of the particular type of vectors, the

Banach space, Norm

A Banach space is a vector space with a norm.

A norm is:

- Non-negative
- Positive definite
- Absolute homogeneous
- Satisfies the triangle inequality

Norm (mathematics) - W

Secure https://en.wikipedia.org/wiki/Norm_(mathematics)

Definition [edit]

Given a vector space V over a subfield F of the complex numbers, a **norm** on V is a fur

For all $a \in F$ and all $\mathbf{u}, \mathbf{v} \in V$,

1. $p(a\mathbf{v}) = |a|\, p(\mathbf{v})$ (being *absolutely homogeneous* or *absolutely scalable*).
2. $p(\mathbf{u} + \mathbf{v}) \leq p(\mathbf{u}) + p(\mathbf{v})$ (being *subadditive* or satisfying the *triangle inequality*).
3. $p(\mathbf{v}) \geq 0$ (being *positive* or more precisely *non-negative*).
4. If $p(\mathbf{v}) = 0$ then $\mathbf{v}=\mathbf{0}$ is the zero vector (being *definite* or being *point-separating*).

There is some redundancy in this definition. By the absolute homogeneity axiom, we ha $p(-\mathbf{v}) + p(\mathbf{v}) \geq p(-\mathbf{v} + \mathbf{v}) = p(\mathbf{0}) = 0$, that is, $p(\mathbf{v}) \geq 0$. Thus, axioms 1 and 2 together imply

A **seminorm** on V is a function $p : V \to \mathbf{R}$ with the properties 1, 2 and 3 above.

Inner Product Space

An inner product space is a Banach space with a dot product (a.k.a. inner product).

An inner product is a function of two vectors that satisfies:

- Conjugate commutative
- Linear in its first argument
- Positive definite

Formally, an inner product space is a vector space V over the field F together with an *inner product*, i.e., v

$$\langle \cdot, \cdot \rangle : V \times V \to F$$

that satisfies the following three axioms for all vectors $x, y, z \in V$ and all scalars $a \in F$:[1][2]

- Conjugate symmetry:[Note 1]

$$\langle x, y \rangle = \overline{\langle y, x \rangle}$$

- Linearity in the first argument:

$$\langle ax, y \rangle = a\langle x, y \rangle$$
$$\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$$

- Positive-definiteness:

$$\langle x, x \rangle \geq 0$$
$$\langle x, x \rangle = 0 \Leftrightarrow x = \mathbf{0}.$$

Alternative definitions, notations and remarks [edit]

Linear Transform

A linear transform converts one vector space into another, with the following rules:

- Homogeneous
- Satisfies superposition

A linear transform is written as a matrix multiplication, $y=Wx$

Definition and first consequences [edit]

Let $\mathbf{V}$ and $\mathbf{W}$ be vector spaces over the same field $\mathbf{K}$. A function $f : \mathbf{V} \to \mathbf{W}$ is said to be a *linear m*
the following two conditions are satisfied:

$f(\mathbf{u}+\mathbf{v}) = f(\mathbf{u}) + f(\mathbf{v})$	additivity / operation of addition
$f(c\mathbf{u}) = cf(\mathbf{u})$	homogeneity of degree 1 / operation of scalar multiplication

Thus, a linear map is said to be *operation preserving*. In other words, it does not matter whether you app
and scalar multiplication.

This is equivalent to requiring the same for any linear combination of vectors, i.e. that for any vectors $\mathbf{u}_1$
equality holds:

$$f(c_1\mathbf{u}_1 + \cdots + c_n\mathbf{u}_n) = c_1 f(\mathbf{u}_1) + \cdots + c_n f(\mathbf{u}_n).$$

[3][4]

Affine Transform

An affine transform is a linear transform plus an offset. It's usually written as $y=Wx+b$, where Wx is the linear part, and b is the offset.

Representation [edit]

As shown above, an affine map is the composition of two functions: a translation and a linear map. Ordinary vector algebra uses matrix multiplication to linear maps, and vector addition to represent translations. Formally, in the finite-dimensional case, if the linear map is represented as a multiplication by and the translation as the addition of a vector $\vec{b}$, an affine map f acting on a vector $\vec{x}$ can be represented as

$$\vec{y} = f(\vec{x}) = A\vec{x} + \vec{b}.$$

Augmented matrix [edit]

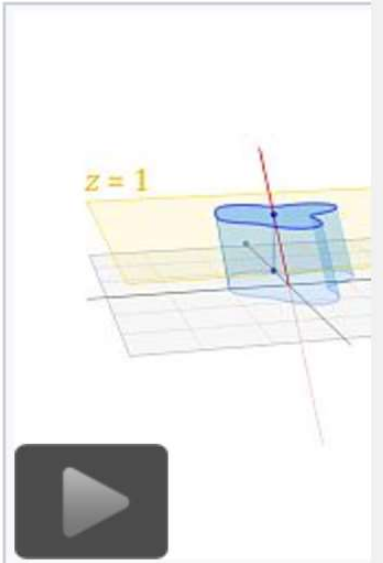
Using an augmented matrix and an augmented vector, it is possible to represent both the translation and the linear map using a single matrix multiplication. The technique requires that all vectors are augmented with a "1" at the end, and all matrices are augmented with an extra row of zeros at the bottom, an extra column—the translation vector—to the right, and a "1" in the lower right corner. If A is a matrix,

$$\begin{bmatrix} \vec{y} \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} & A & & \vec{b} \\ 0 & \dots & 0 & 1 \end{array}\right] \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}$$

is equivalent to the following

$$\vec{y} = A\vec{x} + \vec{b}.$$

The above-mentioned augmented matrix is called an *affine transformation matrix*, or *projective transformation matrix* (as it can also be used to perform projective transformations).



Affine transformations on the

Example Problem

Suppose that $\vec{x}$ is a vector of 312 image features including the color from each of 26 different sub-images (78 features total), and the 3x3 Fourier transform of each of the 26 different sub-images (234 features), for a total of **<u>312 features per image</u>**.

Suppose that $y \in \{-1,1\}$ is its class label (either +1 or -1).

Suppose that $\vec{x}_i$ and y_i are training examples, for $1 \leq i \leq N$.

From these training data, we want to learn a parametric classifier that estimates y from x.

How many trainable parameters?

- Method 1:

$$y = sign(\vec{w}^T x + \vec{b})$$

- Method 2:

$$i = argmax_{i=1}^{N} \, (\vec{x}^T W \vec{x}_i)$$
$$y = sign(y_i)$$

- Method 3:

$$i = argmax_{i=1}^{N} \left(\prod_{d=1}^{D} \frac{1}{\sigma_{di}} e^{-\frac{1}{2}\left(\frac{x_d - \mu_{di}}{\sigma_{di}}\right)^2}\right)$$
$$y = sign(y_i)$$