COMS 647:

Communication Technology

Agenda

- Publishing Internship
- Research Projects
- Review
- Presentation
- Preview

Review

Varieties and Types

Terminology

Collage

Sample

Mashup

Remix

Zeitgeist of the turn of the Century

The remix is the very nature of the digital…The recombinant (the bootleg, the remix, the mash-up) has become the characteristic pivot at the turn of our two centuries (William Gibson, 2005)

Critical Issues

1) For the Record:
The Original Metaphysics of Recording

2) Mashup & Remix:
The Art of Recombinant Rock and Roll

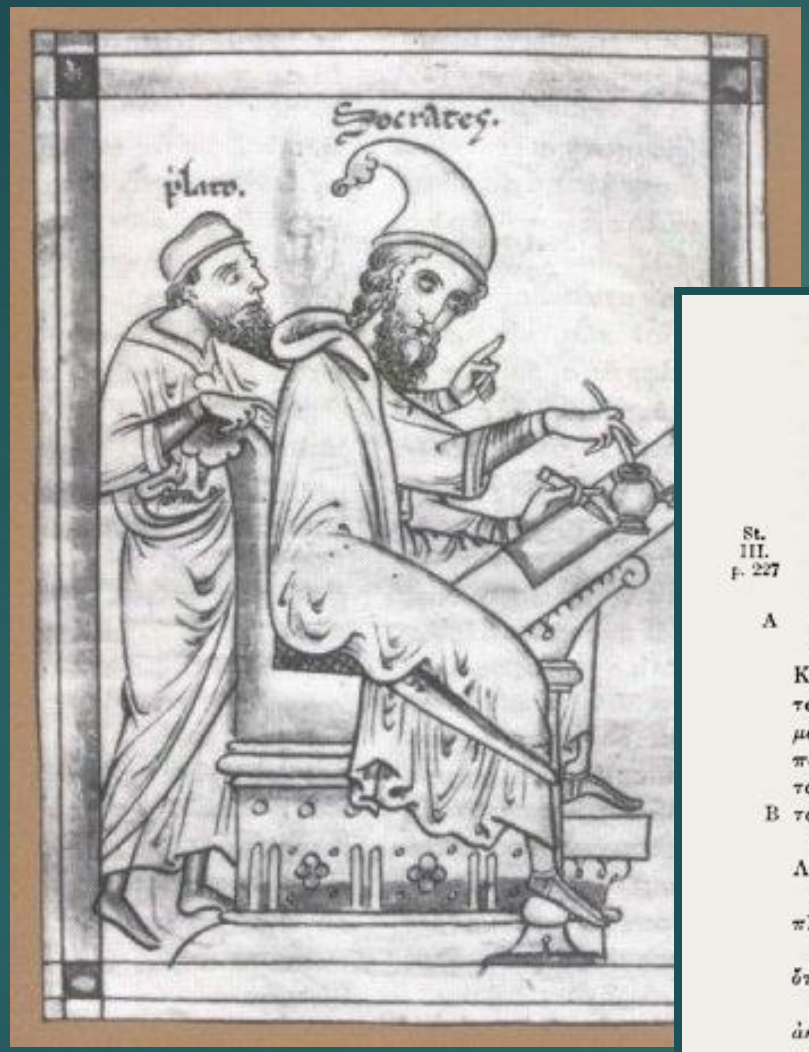
3) Conclusions:
Responses to Remix

1) For the Record:

The Original Metaphysics of Recording

Plato's *Phaedrus*

First recorded account of recording technology



ΦΑΙΔΡΟΣ

[Η ΠΕΡΙ ΚΑΛΟΥ· ΗΘΙΚΟΣ]

St. III. p. 227

ΤΑ ΤΟΥ ΔΙΑΛΟΓΟΥ ΠΡΟΣΩΠΑ

ΣΩΚΡΑΤΗΣ ΚΑΙ ΦΑΙΔΡΟΣ

A 1. ΣΩΚΡΑΤΗΣ. Ὦ φίλε Φαῖδρε, ποῖ δὴ καὶ πόθεν;

ΦΑΙΔΡΟΣ. Παρὰ Λυσίου, ὦ Σώκρατες, τοῦ Κεφάλου· πορεύομαι δὲ πρὸς περίπατον ἔξω τείχους. συχνὸν γὰρ ἐκεῖ διέτριψα χρόνον καθήμενος ἐξ ἑωθινοῦ· τῷ δὲ σῷ καὶ ἐμῷ ἑταίρῳ πειθόμενος Ἀκουμενῷ κατὰ τὰς ὁδοὺς ποιοῦμαι τοὺς περιπάτους· φησὶ γὰρ ἀκοπωτέρους εἶναι

B τῶν ἐν τοῖς δρόμοις.

ΣΩΚΡΑΤΗΣ. Καλῶς γάρ, ὦ ἑταῖρε, λέγει. ἀτὰρ Λυσίας ἦν, ὡς ἔοικεν, ἐν ἄστει.

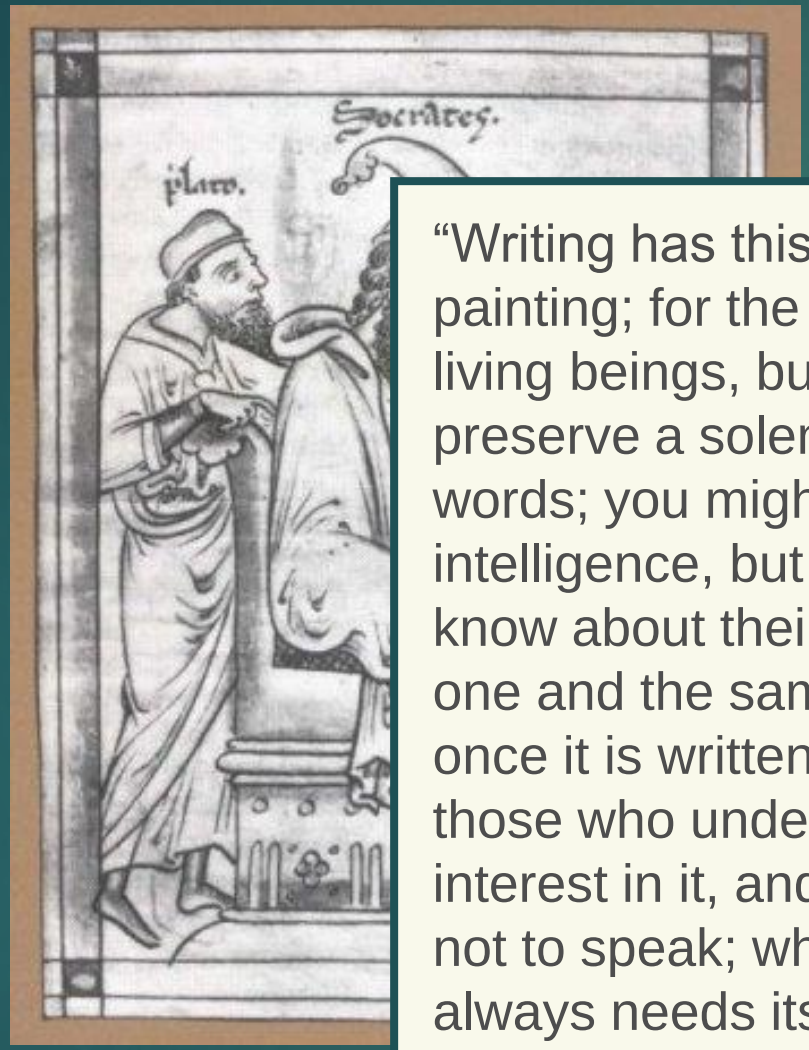
ΦΑΙΔΡΟΣ. Ναί, παρ' Ἐπικράτει, ἐν τῇδε τῇ πλησίον τοῦ Ὀλυμπίου οἰκίᾳ τῇ Μορυχίᾳ.

ΣΩΚΡΑΤΗΣ. Τίς οὖν δὴ ἦν ἡ διατριβή; ἢ δῆλον ὅτι τῶν λόγων ὑμᾶς Λυσίας εἱστία;

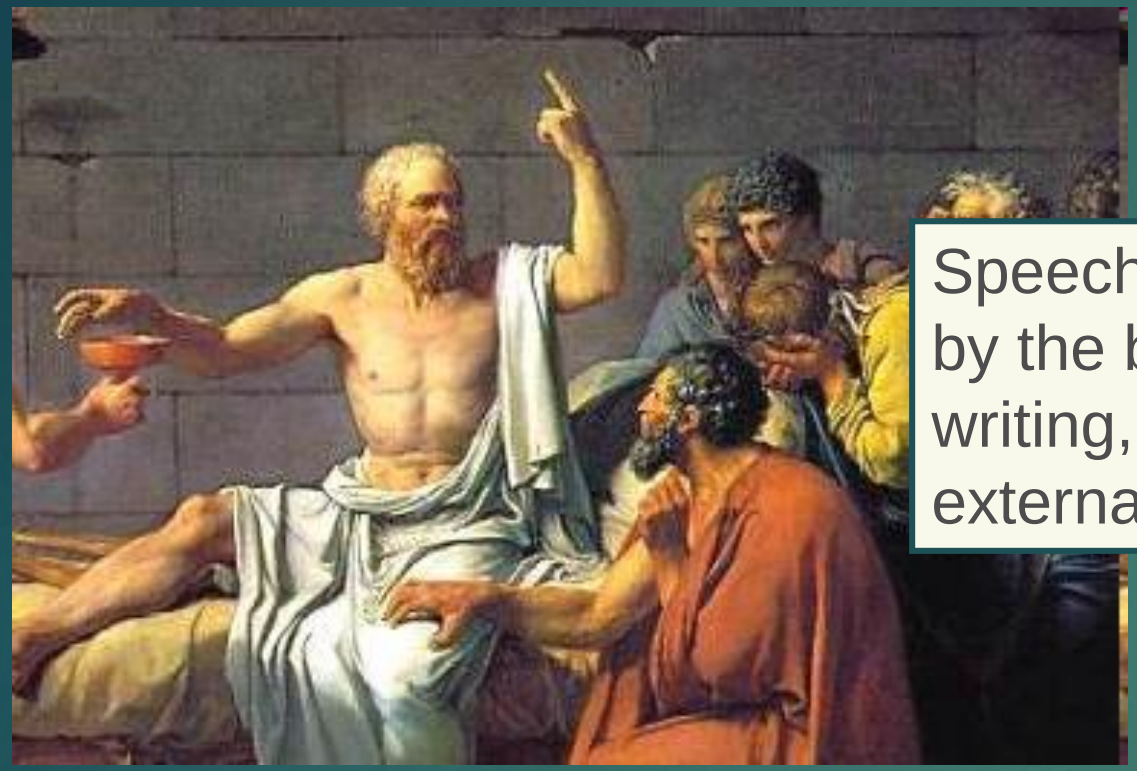
ΦΑΙΔΡΟΣ. Πεύσει, εἴ σοι σχολὴ προϊόντι ἀκούειν.

ΣΩΚΡΑΤΗΣ. Τί δέ; οὐκ ἂν οἶει με κατὰ Πίν-

412



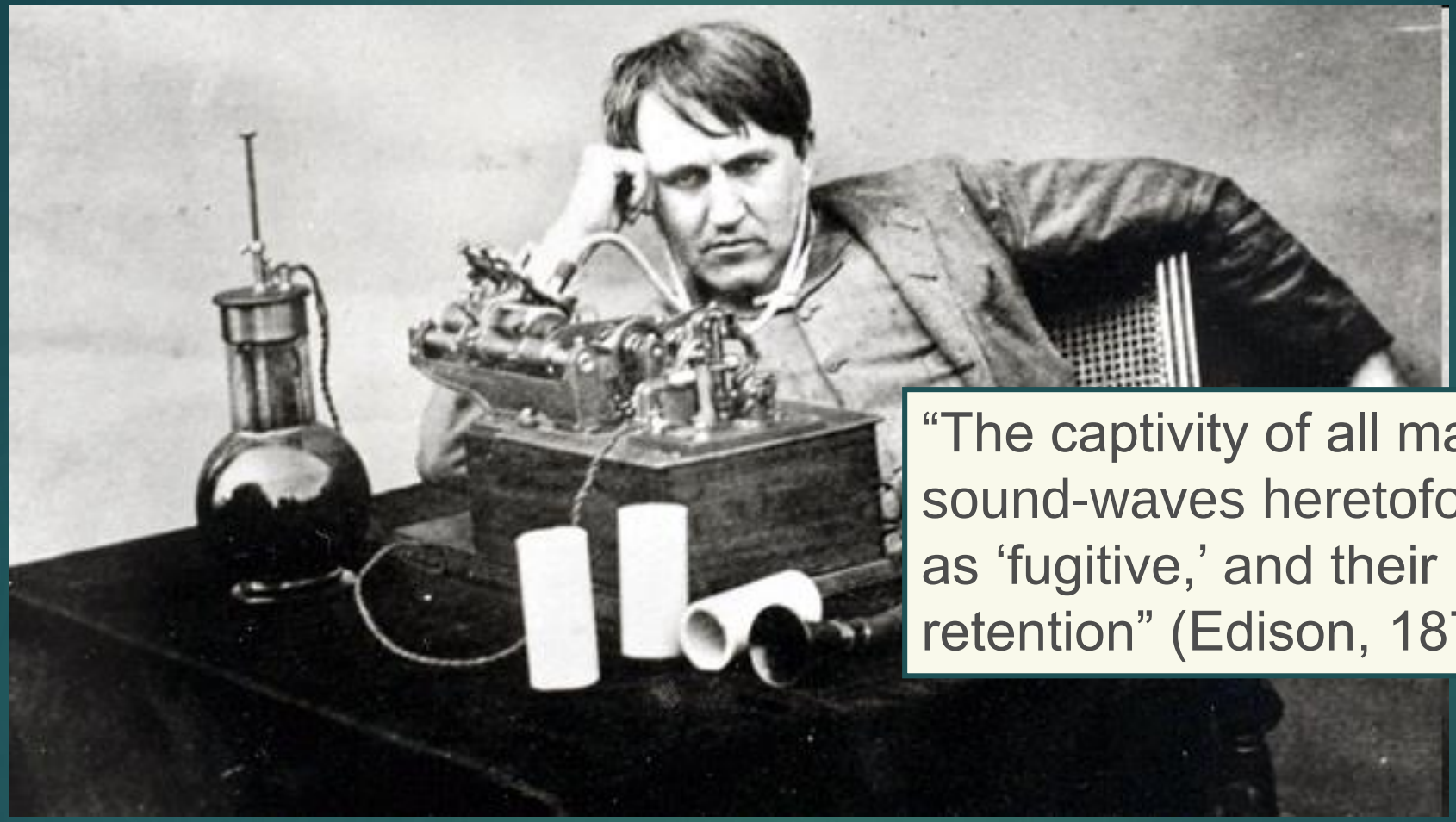
“Writing has this strange quality, and is very like painting; for the creatures of painting stand like living beings, but if one asks them a question, they preserve a solemn silence. And so it is with written words; you might think they spoke as if they had intelligence, but if you question them, wishing to know about their sayings, they always say only one and the same thing. And every word, when once it is written, is bandied about alike among those who understand and those who have no interest in it, and it knows not to whom to speak or not to speak; when ill-treated or unjustly reviled it always needs its father to help it; for it has no power to protect itself” (*Phaedrus* 275d-e).



Speech is alive because it is animated by the breath of a living speaker; while writing, which utilizes artificial and external apparatus, is dead and lifeless.

1. Recordings are secondary and derivative

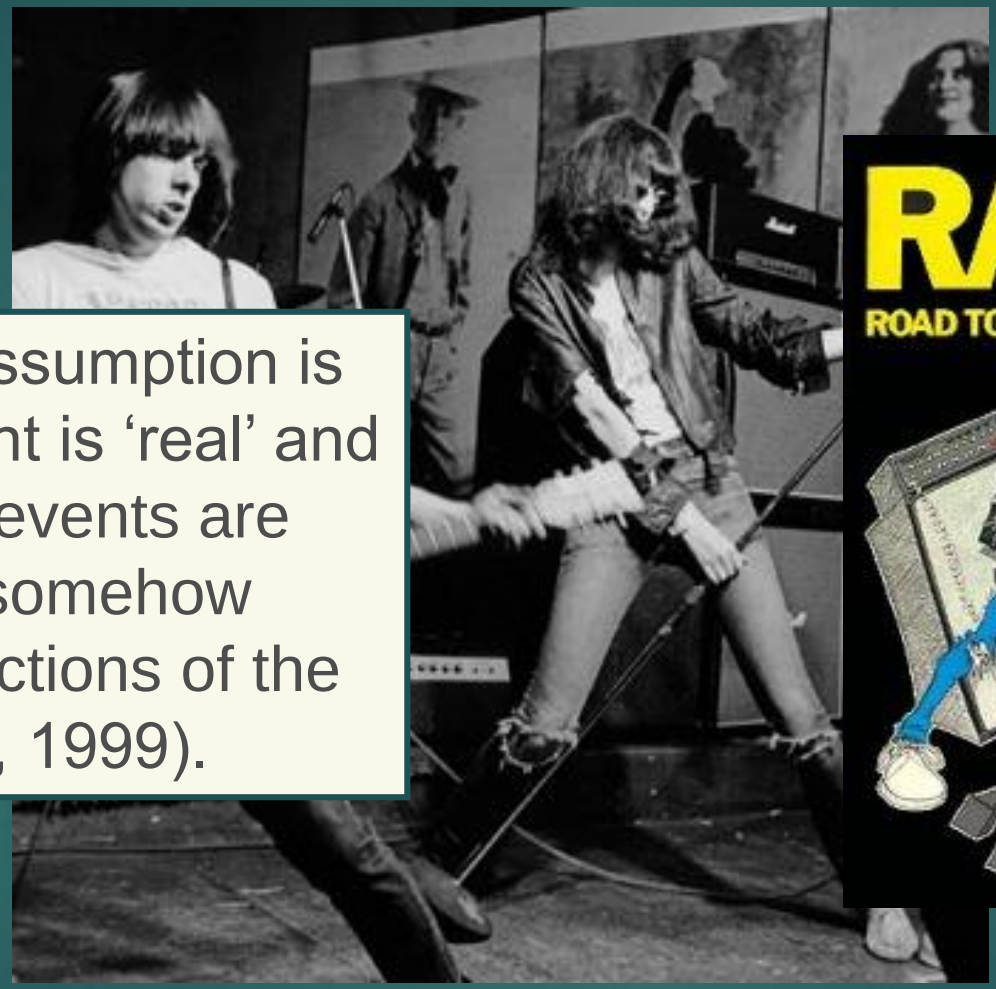
1. Recordings are secondary and derivative

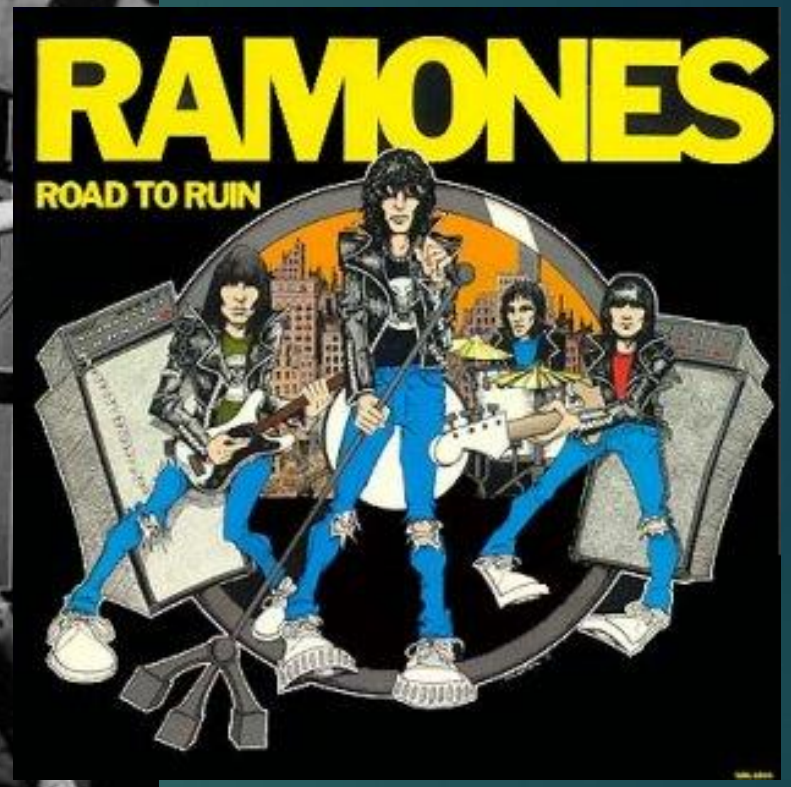


"The captivity of all manner of sound-waves heretofore designated as 'fugitive,' and their permanent retention" (Edison, 1878).

1. Recordings are secondary and derivative

LIVE!





"The common assumption is that the live event is 'real' and that mediatized events are secondary and somehow artificial reproductions of the real" (Auslander, 1999).

1. Recordings are secondary and derivative

2. Recordings preserve live performances

Victor Talking Machine Company

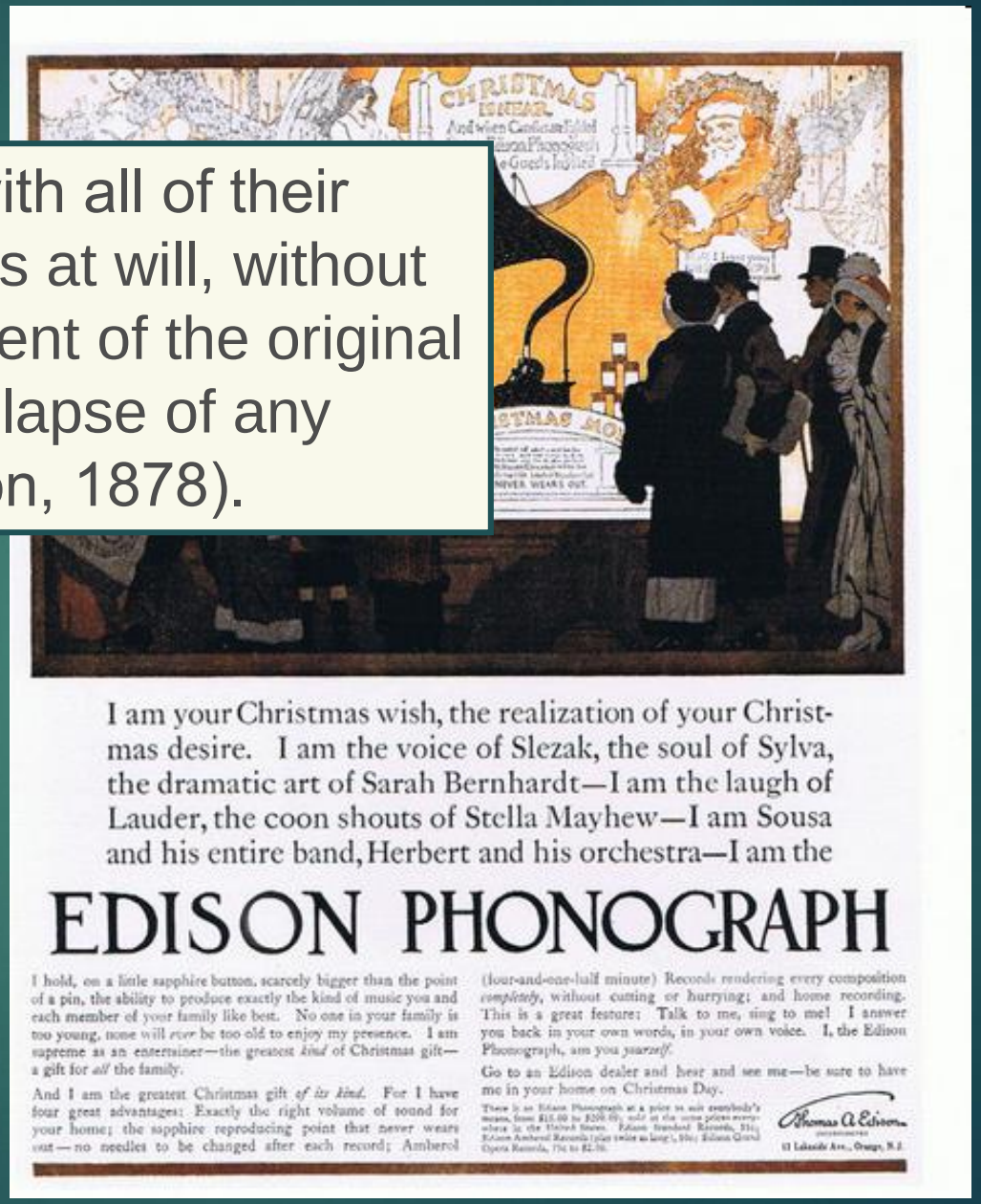
“His Master’s Voice”

2. Recordings preserve live performances

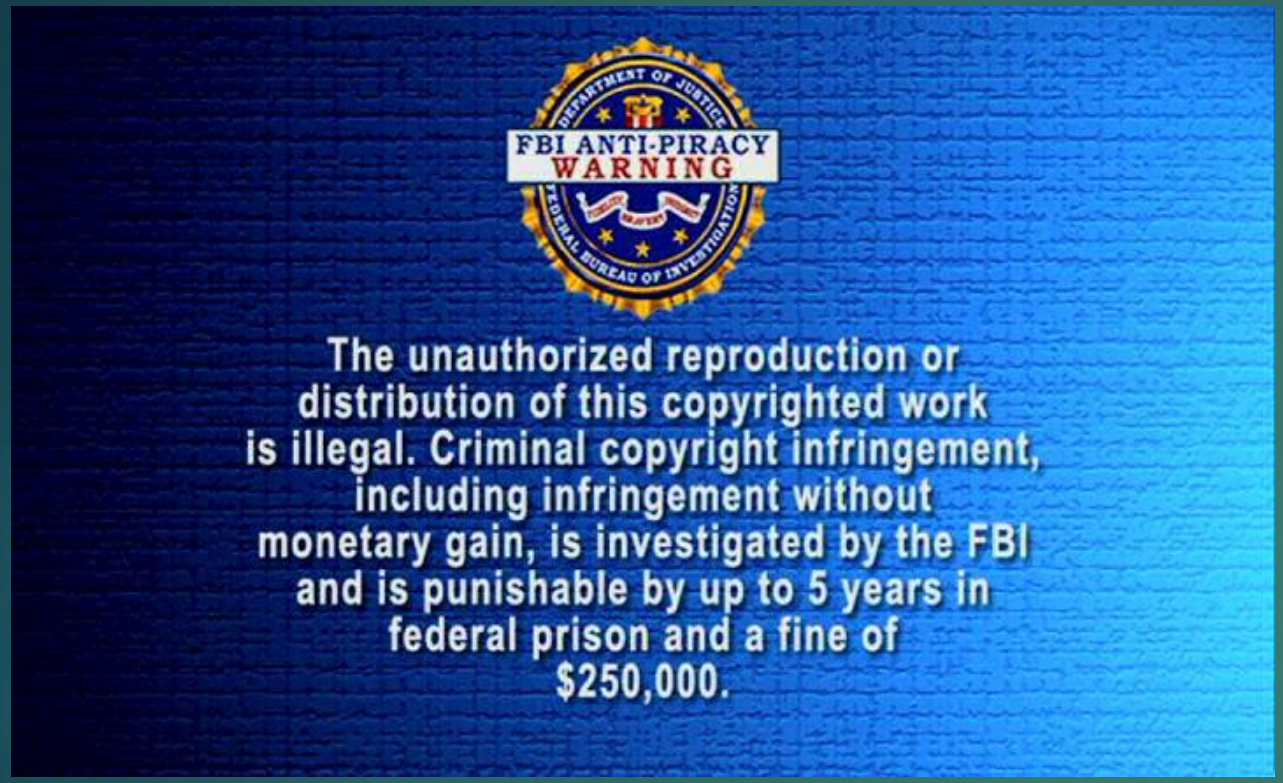
"Music ceases to have interest for us…the instant we become aware of the fact of literal repetition, of mechanical reproduction, when we know and can anticipate exactly how a given phrase is going to be modeled, exactly how long a given fermata is to be held, exactly what quality of accent or articulation, of acceleration, or retard, will occur at a given moment" (Sessions, 1950).

2. Recordings preserve live performances

“Their reproduction with all of their original characteristics at will, without the presence or consent of the original source, and after the lapse of any period of time” (Edison, 1878).



3. Recordings are promiscuous bastards



3. Recordings are promiscuous bastards

3. Recordings are promiscuous bastards

Summary

1) Recordings are secondary and derivative

2) Recordings copy and preserve live performance

3) Recordings are promiscuous bastards

2) Mashup & Remix:

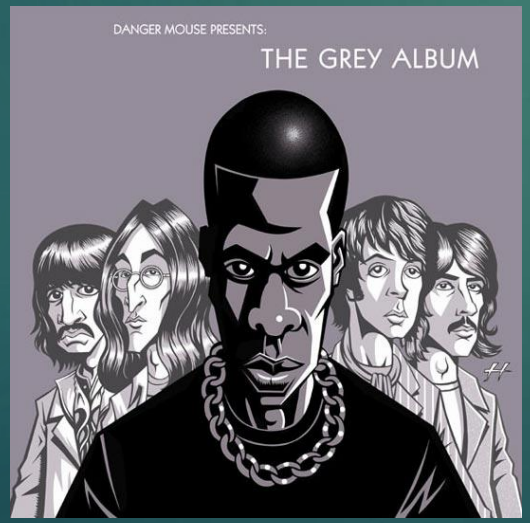
The Art of Recombinant Rock and Roll

1) Mash-ups complicate and suspend common assumptions about origin and originality.

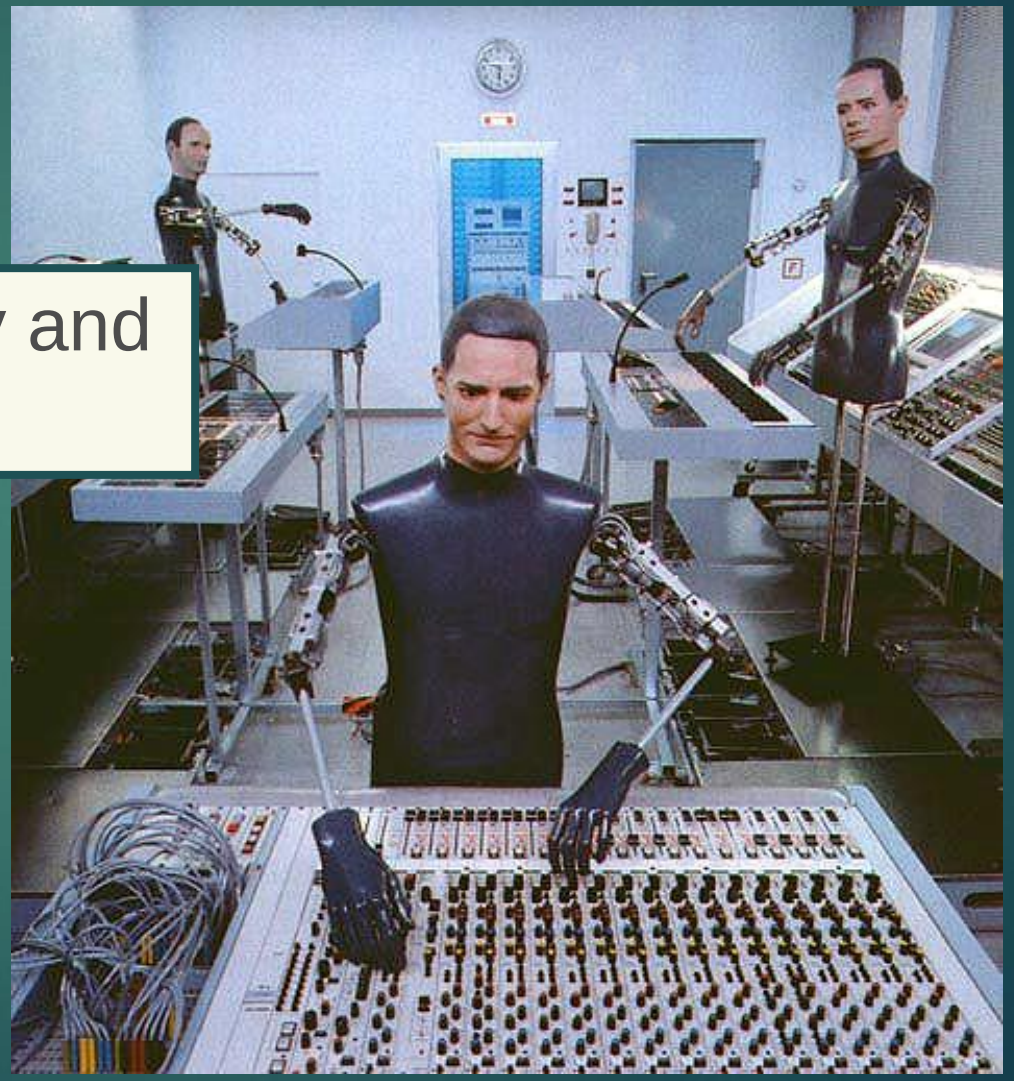
Conceptual Inversion

original

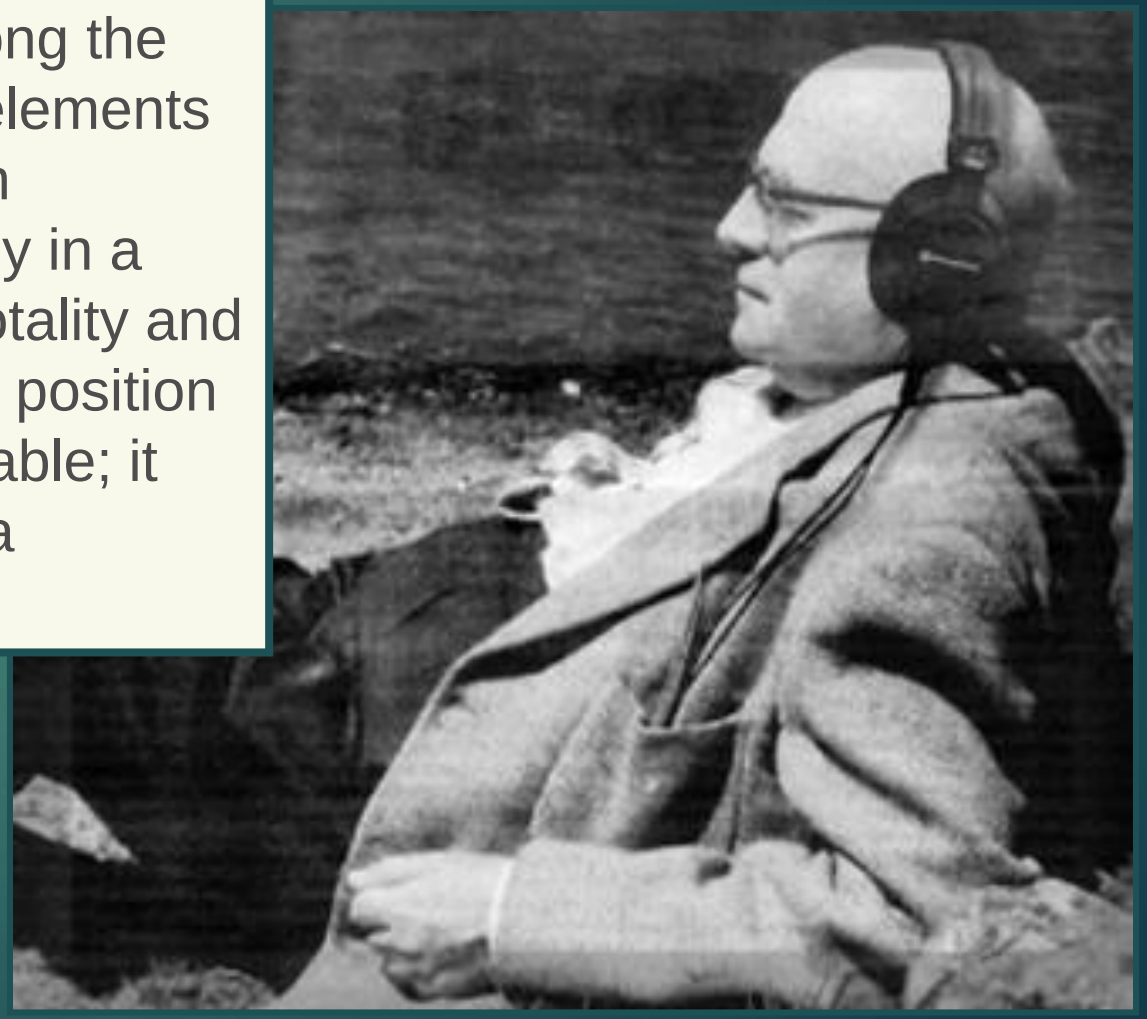
copies



2) Mash-ups are exceedingly and unapologetically redundant

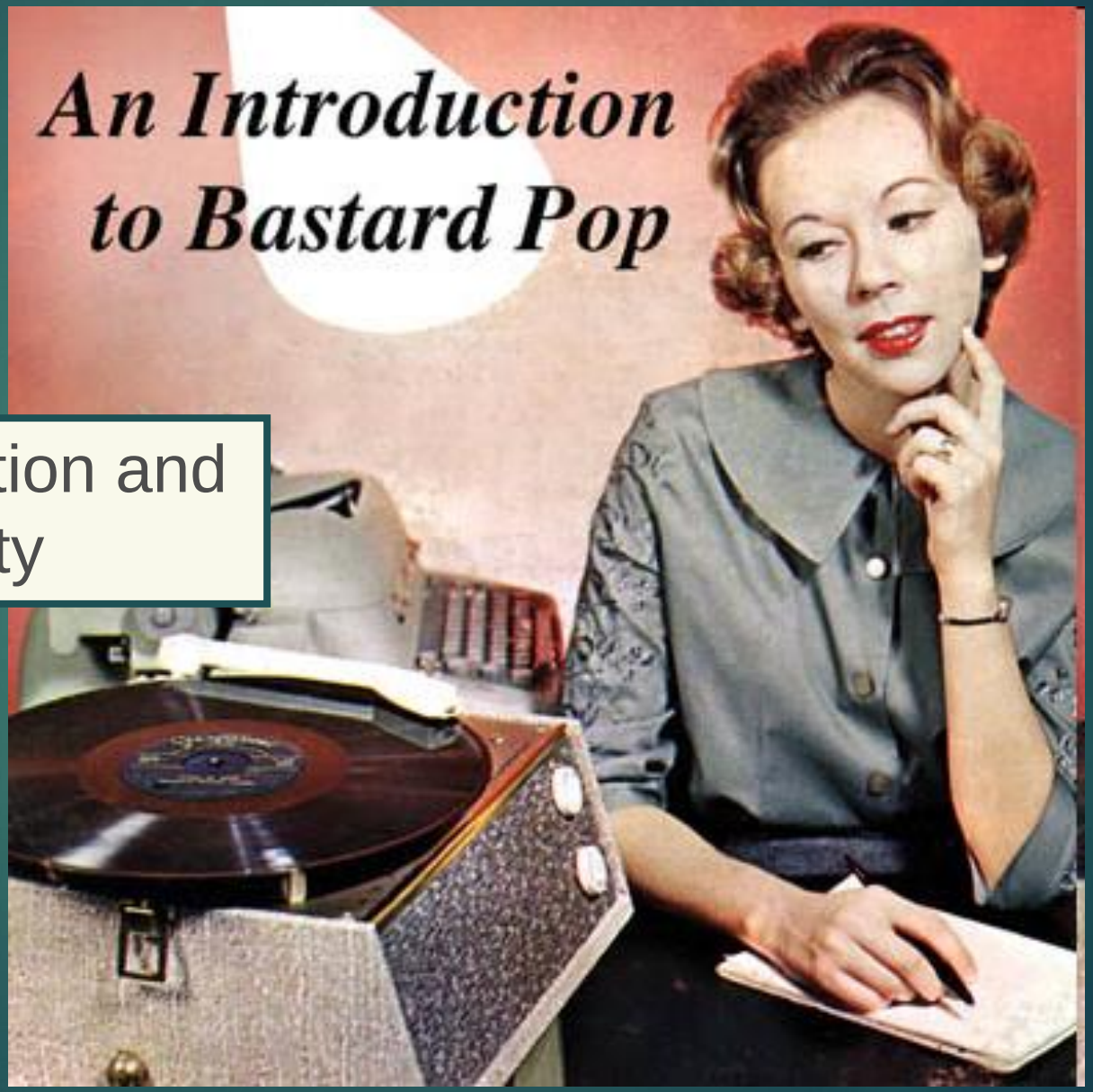


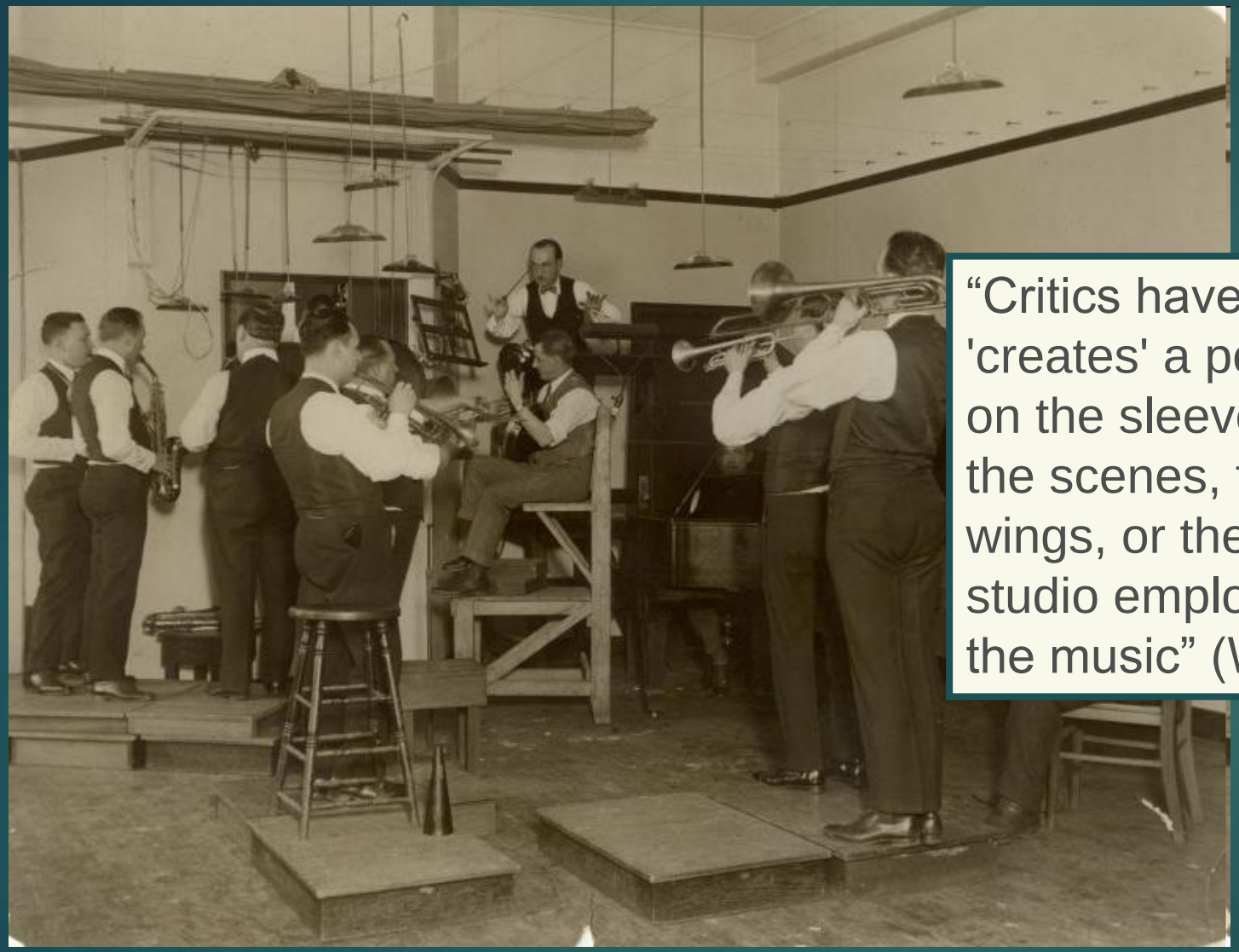
“The beginning of the chorus is replaceable by the beginning of innumerable other choruses. The interrelationship among the elements or the relationship of the elements to the whole would be unaffected. In Beethoven, position is important only in a living relation between a concrete totality and its concrete parts. In popular music, position is absolute. Every detail is substitutable; it serves its function only as a cog in a machine” (Adorno, 1941).




Campbe
CONDENSED
TOMAT
SOUP
GOHOMEPRODUCTIONS
this was pop
2002-2007

3) Mash-ups question and undermine authority





“Critics have long debated who 'creates' a pop record: the artist listed on the sleeve, the producer behind the scenes, the composer in the wings, or the sometimes anonymous studio employees who actually play the music” (Walker, 2003).


You Don't Need a Sword to be a Pirate.
Respect Copyright Laws.
Visit ww2.umflint.edu/piracy to learn how you can ensure that you're on the right side of the law.


Muzik-Knightz.Co.Uk
The Anti-Piracy Division


DEPARTMENT OF JUSTICE
FBI ANTI-PIRACY
WARNING
FIDELITY
BRAVERY
INTEGRITY
BUREAU OF INVESTIGATION

3) Conclusions

Responses to Remix

Two Possible Responses

- Ptolemization
- Copernican Revolution

6.1 Responding to Remix

Responses to these challenges typically take one of two forms, which Žižek, in something of a remix of Thomas Kuhn, calls "Ptolemization" and "Copernican Revolution." "When a discipline is in crisis," Žižek (2008a, vii) explains, "attempts are made to change or supplement its theses *within* the terms of its basic framework—a procedure one might call 'Ptolemization' (since when data poured in which clashed with Ptolemy's earth-centered astronomy, his partisans introduced additional complications to account for the anomalies). But the true 'Copernican' revolution takes place when, instead of just adding complications and changing minor premises, the basic framework itself undergoes a transformation." Ptolemization indicates efforts to revise an existing paradigm by introducing modifications and complications, like the epicycles that were added to the Ptolemaic model to account for seemingly aberrant observational data, in order to ensure the continued functioning and success of the prevailing "normal science." Copernican revolution, on the contrary, designates not minor adjustments or revisions in the prevailing system of knowledge but a complete reconfiguration or transformation of its basic framework. The name, of course, comes from Nicolaus Copernicus, whose heliocentric model of the solar system provides, for Kuhn and others, the prototype of scientific revolution, insofar as it not only introduced a new framework or model of astronomy but literally inverted or overturned the Ptolemaic system by moving the sun, which had been located on the periphery, to the center of the system.

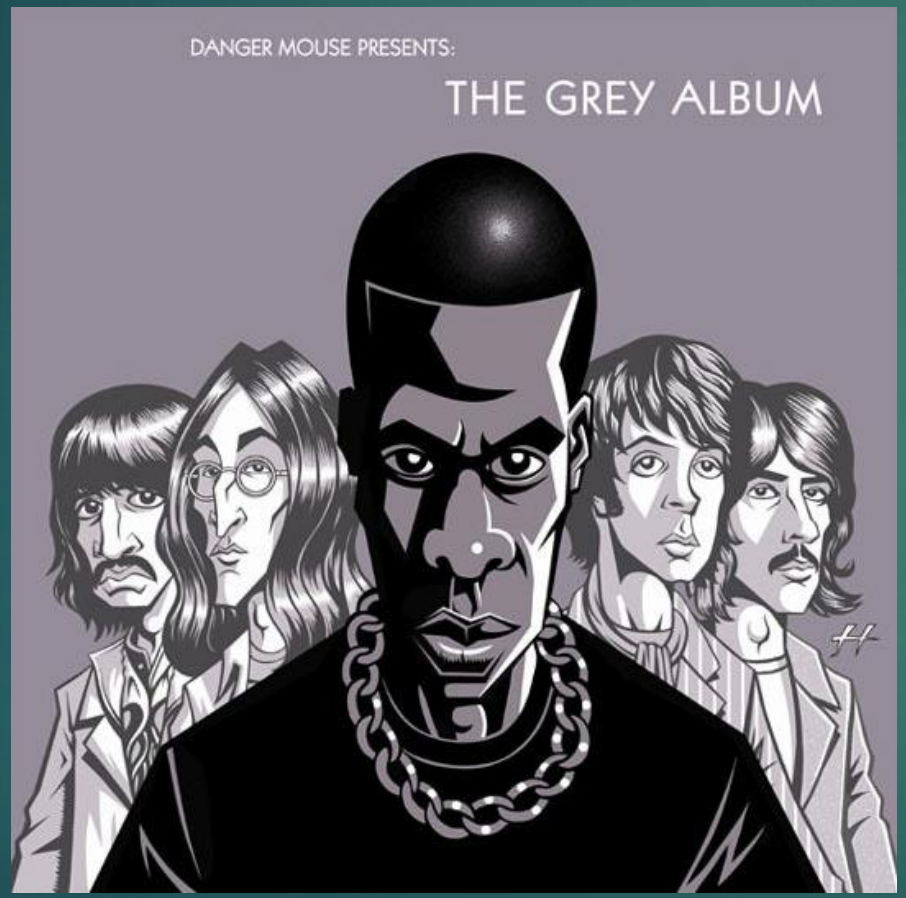
Since remix constitutes something of a crisis in the normal science of recording, it has typically been explained and evaluated in term of the two modes of response described by Žižek's remix of Kuhn. "Does configurable music," as Sinnreich (2010, 193) asks at the end of his analysis, "presage a discursive break in our understanding of what music is and how it operates, and therefore portend a breakdown in the existing social order? Or is the rise of the DJ simply one more development in an ongoing dialectic between cultural regulation and resistance, as easily contained within the strong yet flexible boundaries of the modern framework as atonality, the birth of sound recording, and the electrification of blues and rock music?" Without identifying this source material directly, Sinnreich ends his analysis of the "configurable culture" of remix with a set of questions

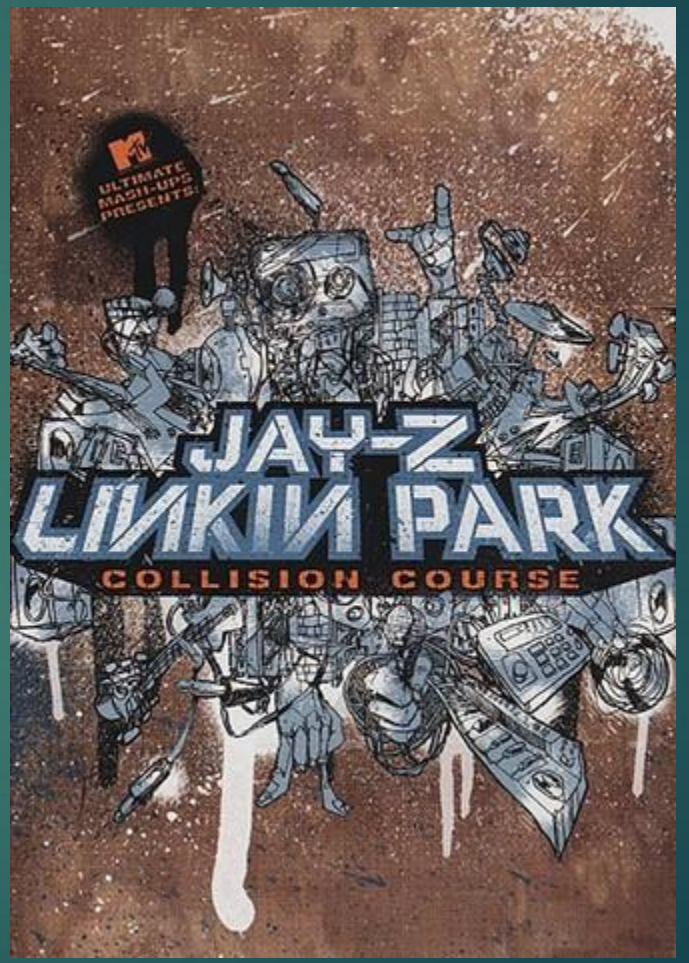
Ptolemization vs. Copernican Revolution

“When a discipline is in crisis attempts are made to change or supplement its theses within the terms of its basic framework—a procedure one might call ‘Ptolemization’ (since when data poured in which clashed with Ptolemy’s earth-centered astronomy, his partisans introduced additional complications to account for the anomalies). But the true ‘Copernican’ revolution takes place when, instead of just adding complications and changing minor premises, the basic framework itself undergoes a transformation.” (Žižek 2008, vii)

Ptolemization

DJ Danger Mouse, *The Grey Album,* 2004





Ptolemization

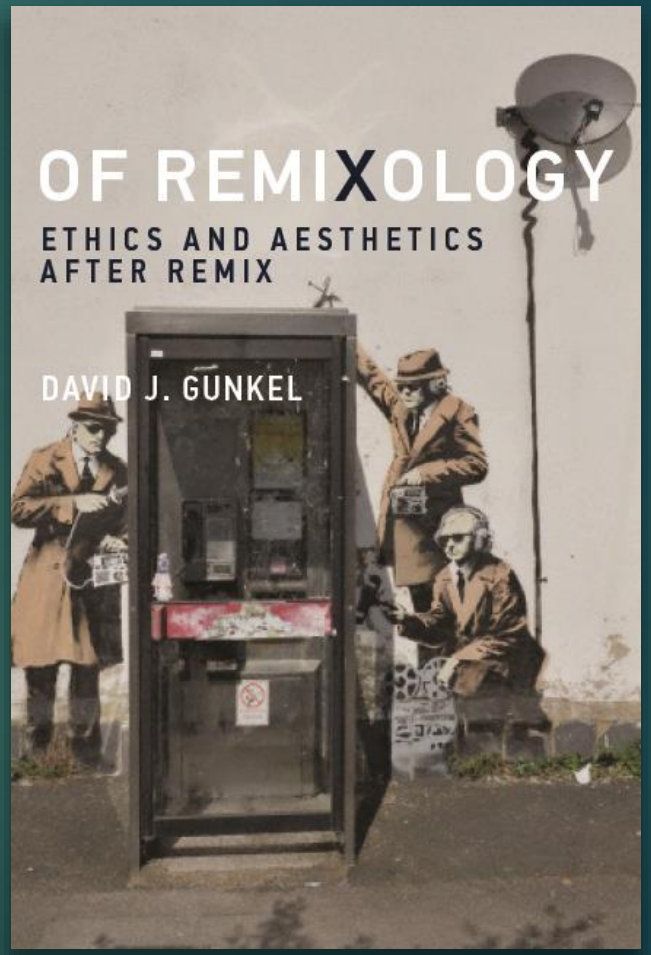
Alexander Rodchenko, *Books*, 1924
"Photomontage"

Copernican Revolution

Copernican Revolution

7.2.1 Competition

The claim of originality, which has been and continues to be a pivotal concept in many aspects of culture (e.g., art, science, commerce), is not itself original. It is always, and from the very beginning, contested and contestable. This contestation, for example, can be seen in the attempt to figure out and decide who created the first mashup. These debates, however, have been inconclusive, and the more we investigate the matter, the more uncertain things seem to become, leaving us not with a single innovator or authoritative voice but with a network of different actors all legitimately claiming some hold on the title of "origination." The trick in this circumstance is not to play according to the rules of the standard Platonic game, which involves trying to distinguish the true claimant—the one true original—from the false pretenders (Deleuze 1990, 254). What is important is not the resolution of the debate but the debate itself. What is valuable and what should be protected, therefore, are not pristine originals but the seemingly unresolvable circulation of things that make dispute over origination possible in the first place.



Copernican Revolution

7.2.1 Competition

The claim of originality, which has been and cont[illegible] cept in many aspects of culture (e.g., art, science, original. It is always, and from the very beginnin[illegible] able. This contestation, for example, can be seen out and decide who created the first mashup. have been inconclusive, and the more we investig uncertain things seem to become, leaving us not or authoritative voice but with a network of differ claiming some hold on the title of "origination." stance is not to play according to the rules of the which involves trying to distinguish the true clai nal—from the false pretenders (Deleuze 1990, 254). What is important is not the resolution of the debate but the debate itself. What is valuable and what should be protected, therefore, are not pristine originals but the seemingly unresolvable circulation of things that make dispute over origination possible in the first place.

7.2.2 Decision

These competing claims can only be settled—even now at a time when one might believe that things operate otherwise—by an arbitrary and even random selection, or as Deleuze (1994, 62–63) characterizes it, by way of a grounding that is ungrounded. The question concerning origination, therefore, is resolved on the basis of a socially constructed decision, quite literally a cut made in the maelstrom of the eternal recurrence, that arbitrarily (although not without reason, argumentation, and supporting evidence) extracts and promotes one claimant as the originator. "Originality,"

Copernican Revolution

7.2.1 Competition

The claim of originality, which has been and cont
cept in many aspects of culture (e.g., art, science,
original. It is always, and from the very beginnin
able. This contestation, for example, can be seen
out and decide who created the first mashup.
have been inconclusive, and the more we investi
uncertain things seem t
or authoritative voice bu
claiming some hold on
stance is not to play acc
which involves trying to
nal—from the false pret
not the resolution of the
what should be protecte
ingly unresolvable circul
possible in the first place

7.2.2 Decision

These competing claims can only be settled—even now at a time when one might believe that things operate otherwise—by an arbitrary and even random selection, or as Deleuze (1994, 62–63) characterizes it, by way of a grounding that is ungrounded. The question concerning origination,
cted decision, quite
currence, that arbi-
and supporting evi-
nator. "Originality,"

7.2.3 Finitude

Because a decision concerning the assignment of origination is contingent and socially constructed, it should have a predetermined shelf life or expiration date. The "original," or better, the contingent decision settling a claim to originality, should be temporary and finite. Following (but also reconfiguring) the precedent of US patent law in pharmaceutical development (and by using an analogy to drugs, we come full circle to the pharmacology originally deployed in Plato's *Phaedrus*), the claim to originality should be granted for only a limited and rather short period of time (three to five years, for example), after which the claim expires, the protections accorded the so-called original no longer hold, and the "innovation" becomes part of the general fabric of culture—in the parlance of US copyright law, "public domain"—and therefore available for further sampling, remixing, and mashing up. This restricted period of time would allow successful claimants to capitalize on the efforts and investments they made in the process of developing an "innovation" while simultaneously recognizing that any "innovation" in any field of endeavor is derived from the work of others and must therefore give back in kind, becoming source material for future efforts.

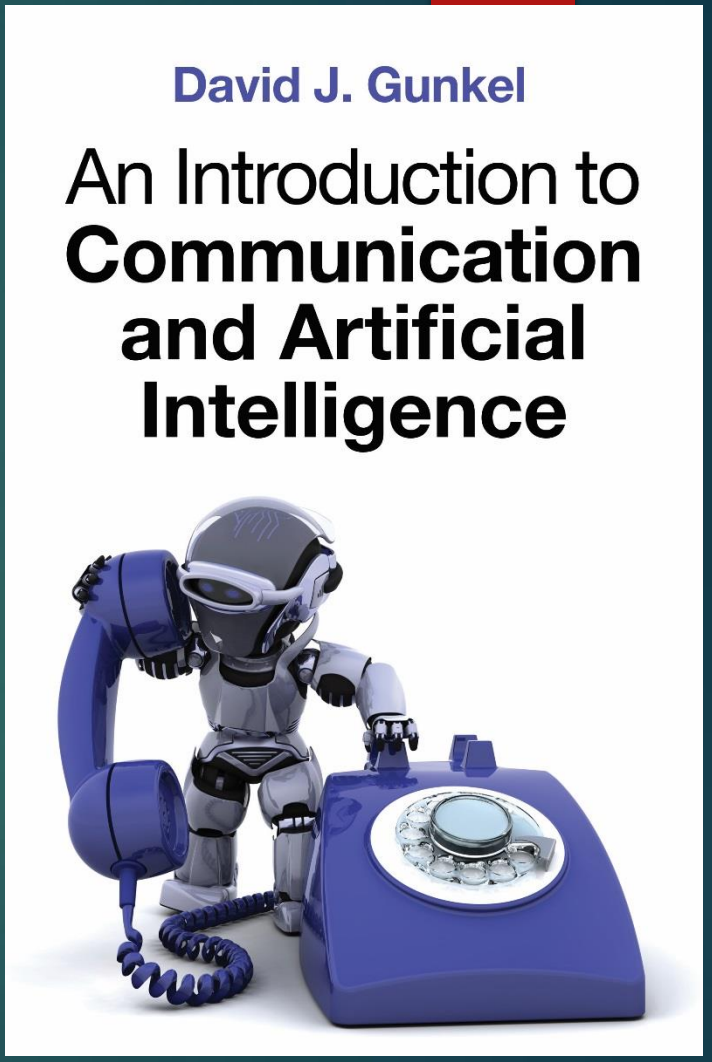
Remix Spectrum

Copernican Revolution

Ptolemization

Today

- Artificial Intelligence
  - Intro to Communication & AI - ch.1-3
  - PBS - The Chinese Room (video)
  - Steiner - Algorithms Are Taking Over (video)

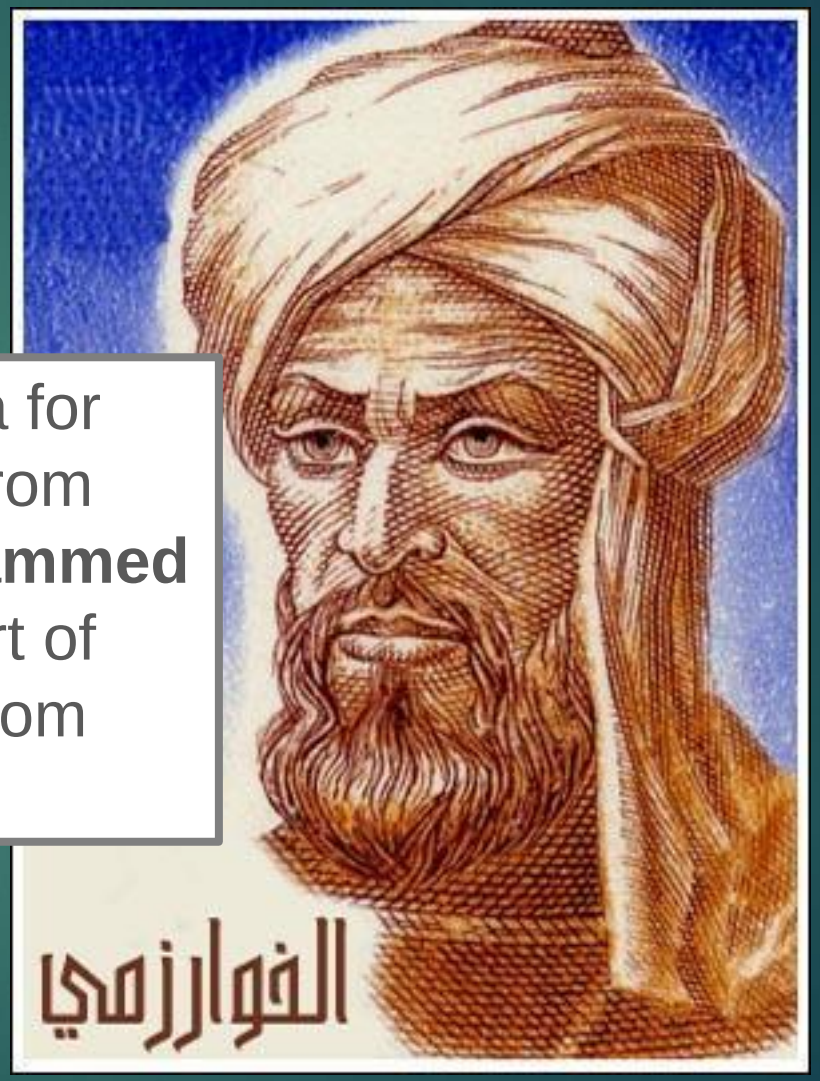


Algorithm Exercise

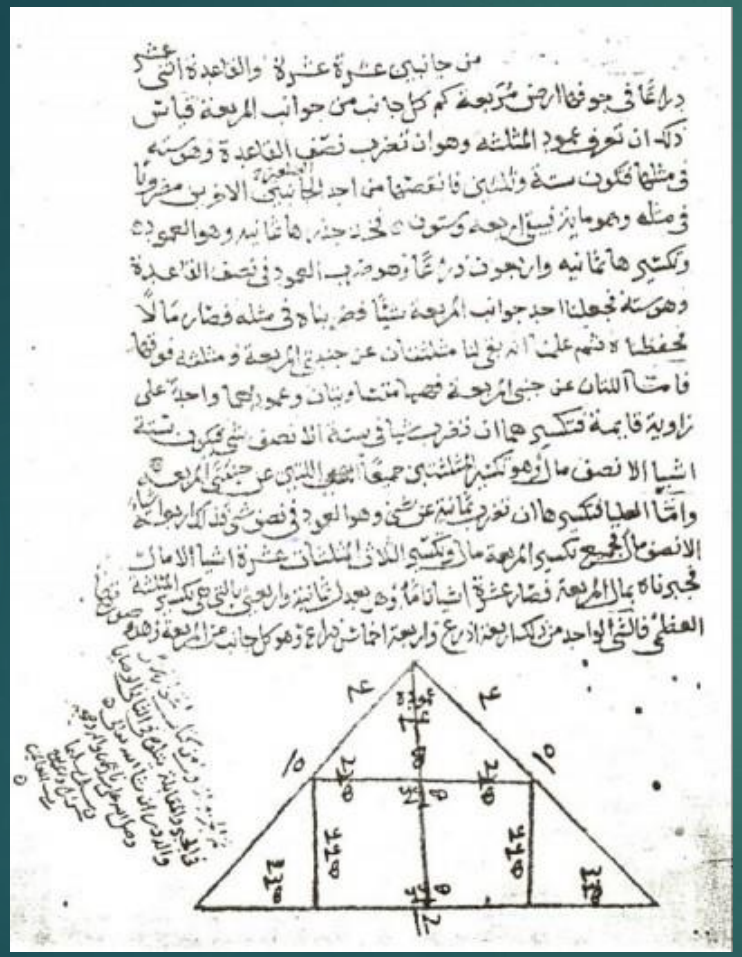
- Introduction
- Maker Exercise – Temperature Converter
  - GOFAI Algorithm
  - Machine Learning Algorithm

Algorithm Exercise

An algorithm is a procedure or formula for solving a problem. The word derives from the name of the mathematician, **Mohammed ibn-Musa al-Khwarizmi**, who was part of the royal court in Baghdad and lived from about 780 to 850.



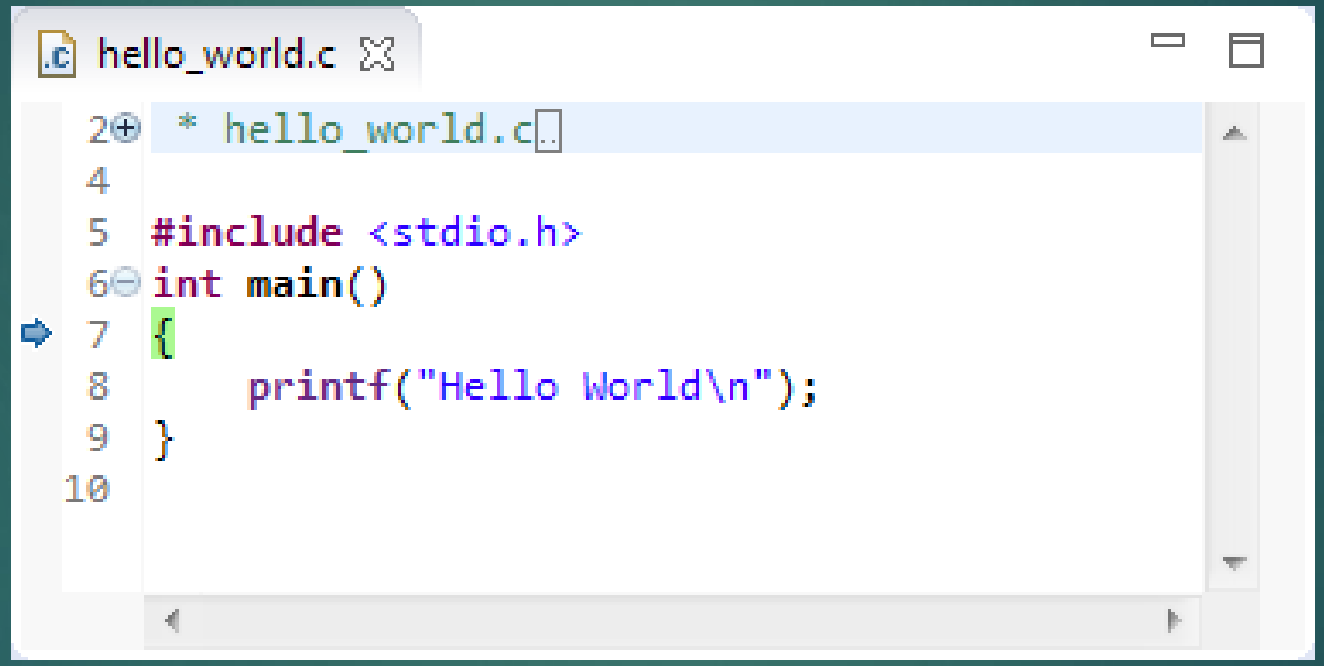
Algorithm Exercise



Directions

1. Wet hair and lather.
2. Rinse thoroughly.
3. Repeat if necessary.

Algorithm Exercise



Algorithm Exercise

Definition

Javascript is a light-weight, high-level, dynamic, untyped, and interpreted programming language.

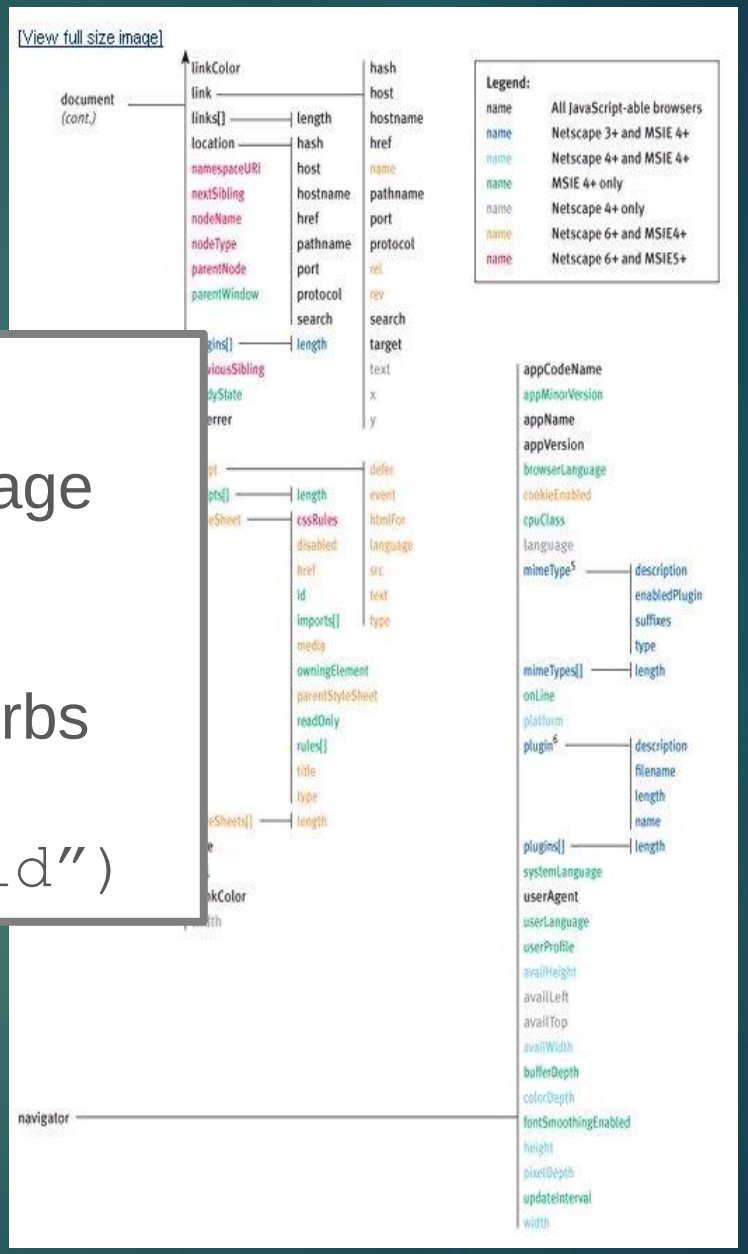
Translation: Javascript is a powerful but easy to use and learn programming language for writing web applications.

Algorithm Exercise

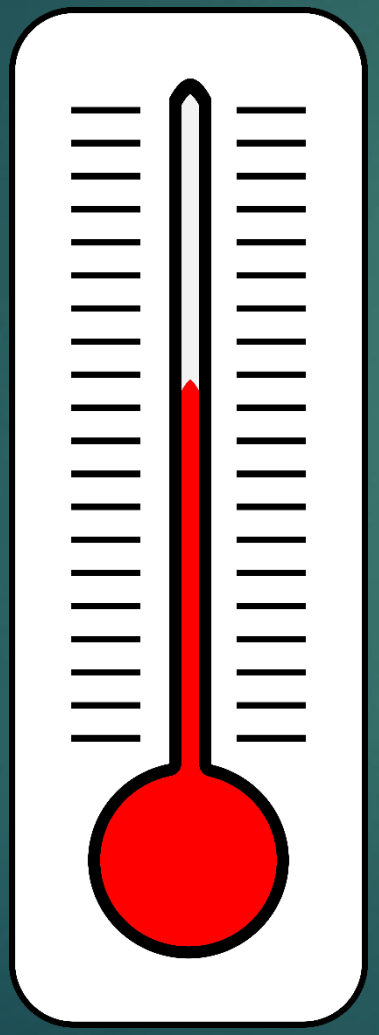
Features

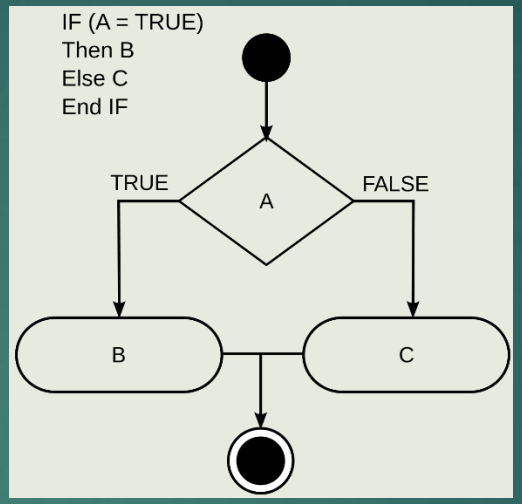
- Object Oriented Programming Language
  - Objects = nouns
  - Methods = verbs
  - Properties = adjectives and adverbs
- Dot Syntax (sentence structure)

```
document.write("hello world")
```

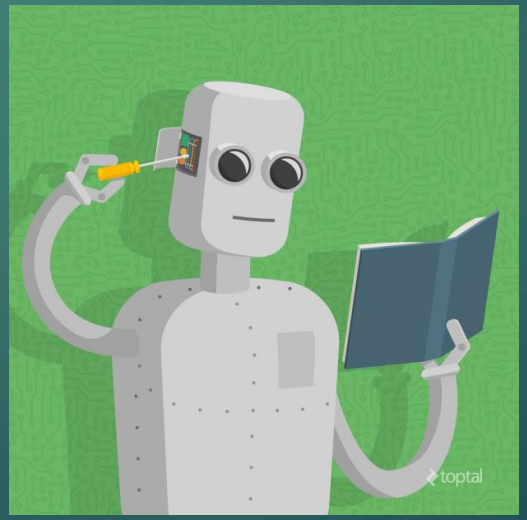


Algorithm Exercise



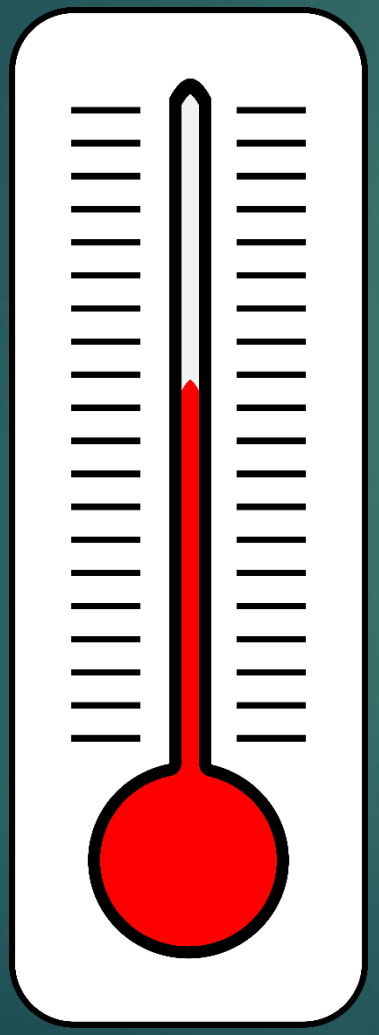


Temp Conversion – v1
Symbolic Reasoning
GOFAI approach



Temp Conversion – v2
Machine Learning
Neural Network

v1 – GOFAI



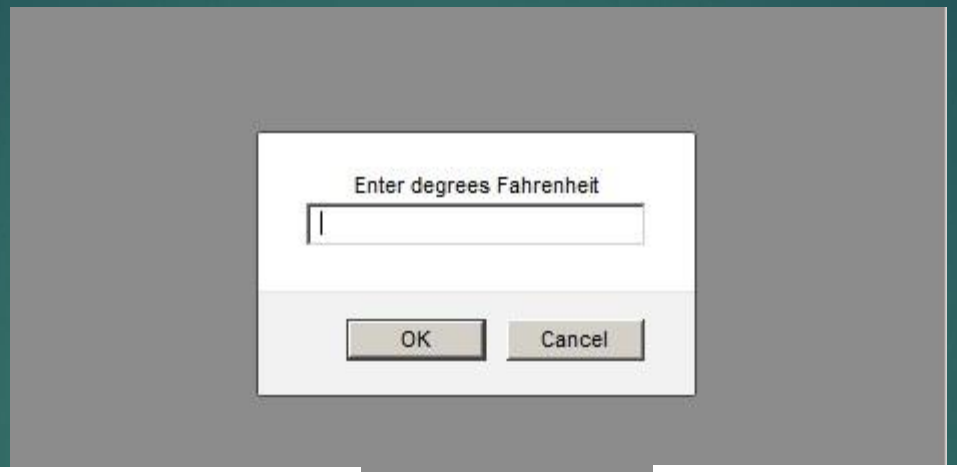
Overview

- Demonstrate rule-based, symbolic reasoning approach or GOFAI
- Use coded instructions to tell the computer step-by-step what to do in order to make the temperature conversion
- Three versions – Start small and then add incrementally to the code to make the temperature converter more accurate and robust

temp-version1.html

```
1
2   <script>
3
4     var TempF = prompt("Enter degrees Fahrenheit");
5     var TempC;
6
7     if(TempF == 32) TempC = 0;
8     else TempC = "UNDEFINED";
9
10    document.write("<h1>" + TempF + " converts to " + TempC + "</h1>");
11
12  </script>
13
```

temp-version1.html



32 converts to 0

50 converts to UNDEFINED

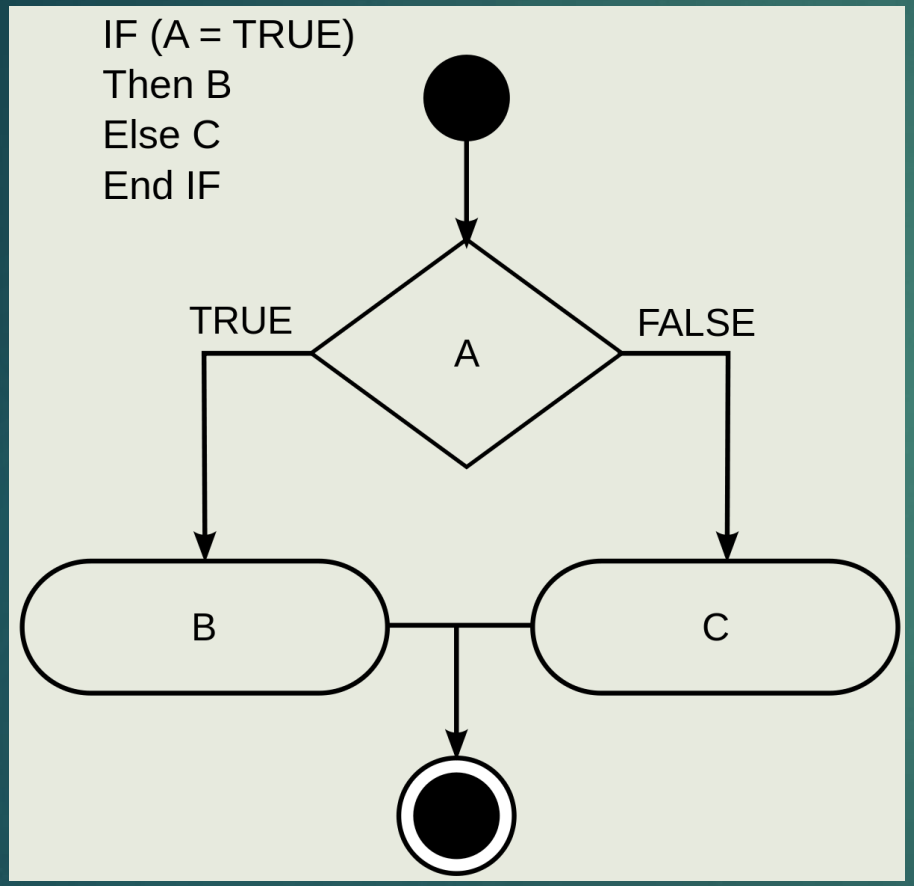
Modification #1

```
1
2   <script>
3
4     var TempF = prompt("Enter degrees Fahrenheit");
5     var TempC;
6
7     if(TempF == 32) TempC = 0;
8
9     else if(TempF == 50) TempC = 10;
10
11    else if(TempF ==99) TempC = 37.2;
12
13    else TempC = "UNDEFINED";
14
15    document.write("<h1>" + TempF + " converts to " + TempC + "</h1>");
16
17  </script>
18
```

Modification #2

```
1
2   <script>
3
4     var TempF = prompt("Enter degrees Fahrenheit");
5     var TempC;
6
7     if(TempF == 32) TempC = 0;
8
9     else if(TempF > 32 && TempF < 49) TempC = 4;
10
11    else if(TempF == 50) TempC = 10;
12
13    else if(TempF > 50 && TempF < 98) TempC = 21;
14
15    else if(TempF ==99) TempC = 37.2;
16
17    else TempC = "UNDEFINED";
18
19    document.write("<h1>" + TempF + " converts to " + TempC + "</h1>");
20
21  </script>
22
```

v1 – GOFAI



Summary

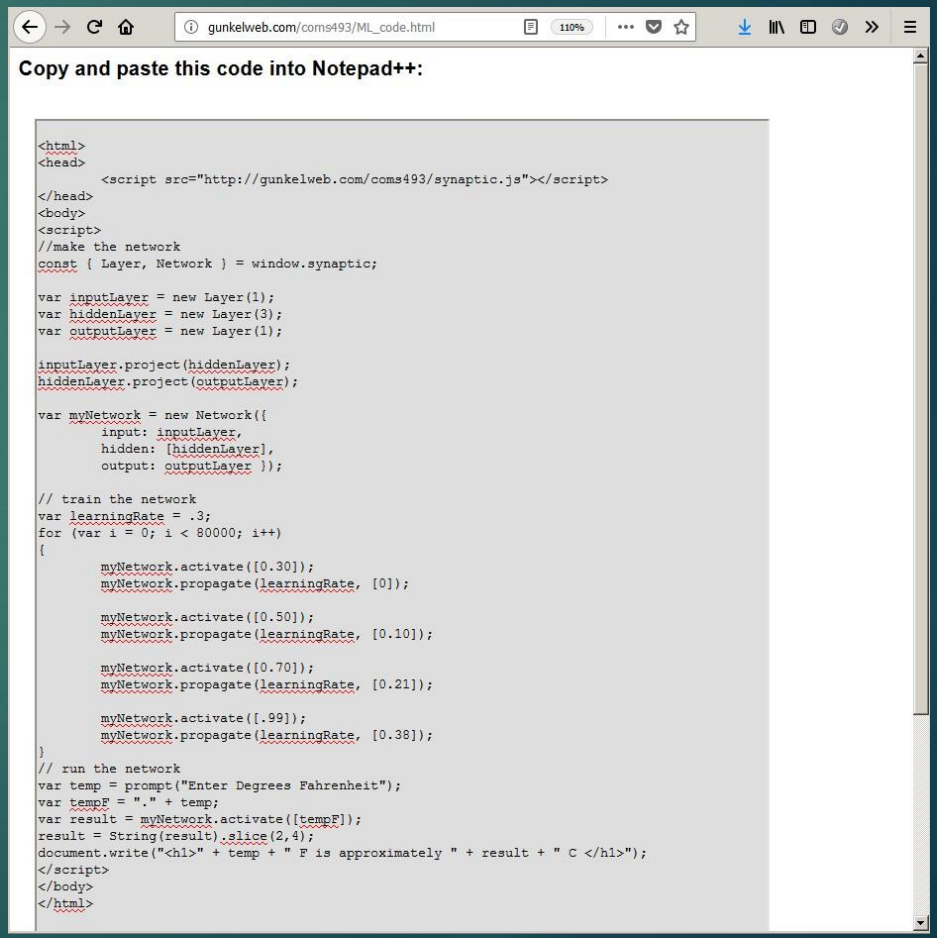
- Step-by-Step instructions
- Programmer must know the temperature conversions (i.e. 32 F = 0 C)
- Encode the conversion steps in Javascript
- More accurate converter = More lines of coded instructions

v2 – Machine Learning

http://gunkelweb.com/coms493/ML_code.html

Directions

1. Go to website
2. Copy the text in the grey box
3. Paste this copied text into Notepad++
4. Save as `temp_version2.html`
5. Open the file in the browser and try it



temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

1. Load the Synaptic Library

"Synaptic is a javascript neural network library. Its generalized algorithm is architecture-free, so you can build and train basically any type of first order or even second order neural network architectures."

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/s
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approxima
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

https://cdnjs.cloudflare.com/ajax/libs/synaptic/1.0.10/synaptic.js

```
/*!
 * The MIT License (MIT)
 *
 * Copyright (c) 2017 Juan Cazala - juancazala.com
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
 * THE SOFTWARE
 *
 *
 *
 * ********************************************************************************************
 *                                      SYNAPTIC (v1.0.10)
 * ********************************************************************************************
 *
 * Synaptic is a javascript neural network library for node.js and the browser, its generalized
 * algorithm is architecture-free, so you can build and train basically any type of first order
 * or even second order neural network architectures.
 *
 * http://en.wikipedia.org/wiki/Recurrent_neural_network#Second_Order_Recurrent_Neural_Network
 *
 * The library includes a few built-in architectures like multilayer perceptrons, multilayer
 * long-short term memory networks (LSTM) or liquid state machines, and a trainer capable of
 * training any given network, and includes built-in training tasks/tests like solving an XOR,
 * passing a Distracted Sequence Recall test or an Embeded Reber Grammar test.
 *
 * The algorithm implemented by this library has been taken from Derek D. Monner's paper:
 *
 *
 * A generalized LSTM-like training algorithm for second-order recurrent neural networks
 * http://www.overcomplete.net/papers/nn2012.pdf
 *
 * There are references to the equations in that paper commented through the source code.
 *
 */
```

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/s
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approxima
</script>
</body>
</html>
```

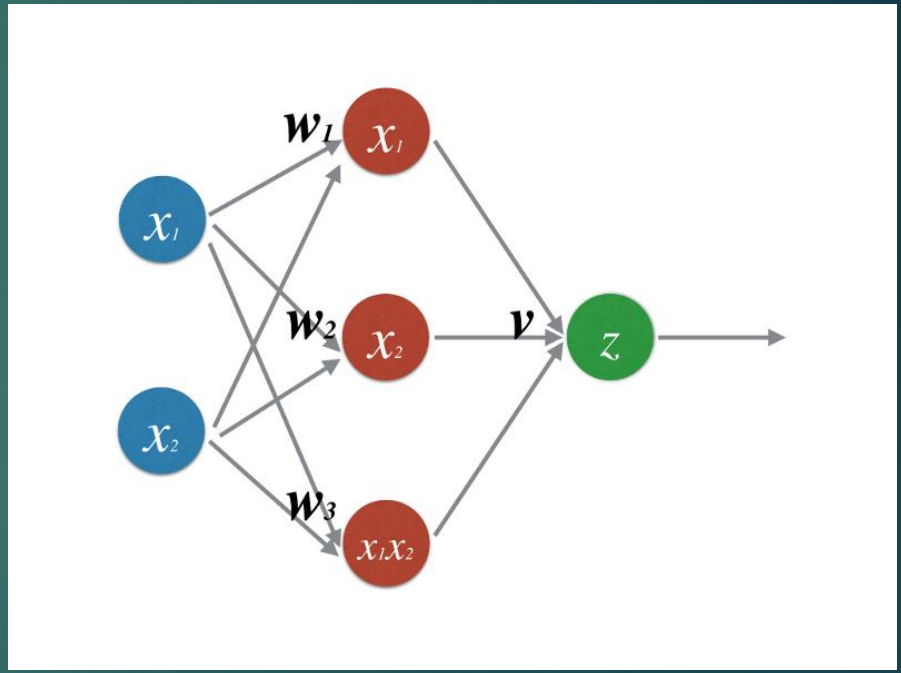
```
/*******************************************************************************************
                                        NEURON
*******************************************************************************************/

function Neuron() {
  this.ID = Neuron.uid();
  this.label = null;
  this.connections = {
    inputs: {},
    projected: {},
    gated: {}
  };
  this.error = {
    responsibility: 0,
    projected: 0,
    gated: 0
  };
  this.trace = {
    elegibility: {},
    extended: {},
    influences: {}
  };
  this.state = 0;
  this.old = 0;
  this.activation = 0;
  this.selfconnection = new Neuron.connection(this, this, 0); // weight = 0 -> not connected
  this.squash = Neuron.squash.LOGISTIC;
  this.neighboors = {};
  this.bias = Math.random() * .2 - .1;
}

Neuron.prototype = {

  // activate the neuron
  activate: function(input) {
    // activation from enviroment (for input neurons)
    if (typeof input != 'undefined') {
      this.activation = input;
      this.derivative = 0;
      this.bias = 0;
      return this.activation;
    }

    // old state
    this.old = this.state;

    // eq. 15
    this.state = this.selfconnection.gain * this.selfconnection.weight *
      this.state + this.bias;

    for (var i in this.connections.inputs) {
      var input = this.connections.inputs[i];
      this.state += input.from.activation * input.weight * input.gain;
    }
```

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

2. Construct the Neural Network



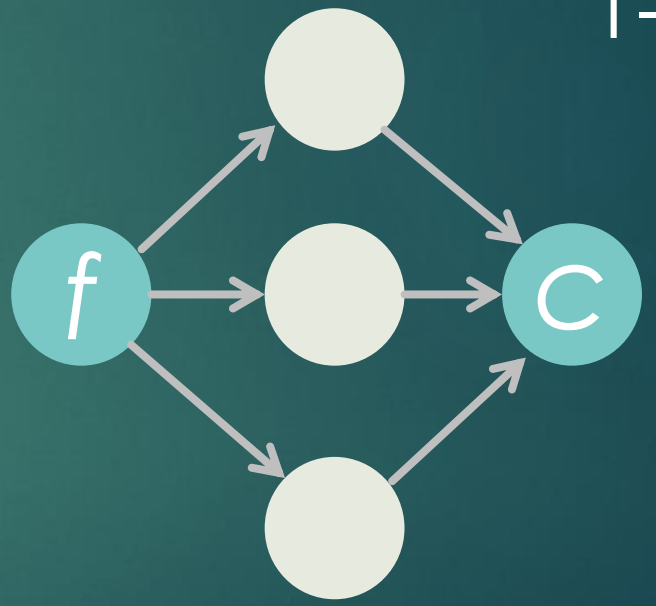
temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

2. Construct the Neural Network

1-3-1



Input Layer

Hidden Layer

Output Layer

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

3. Train the Network on Data

Four pieces of data:

30 F / 0 C

50 F / 10 C

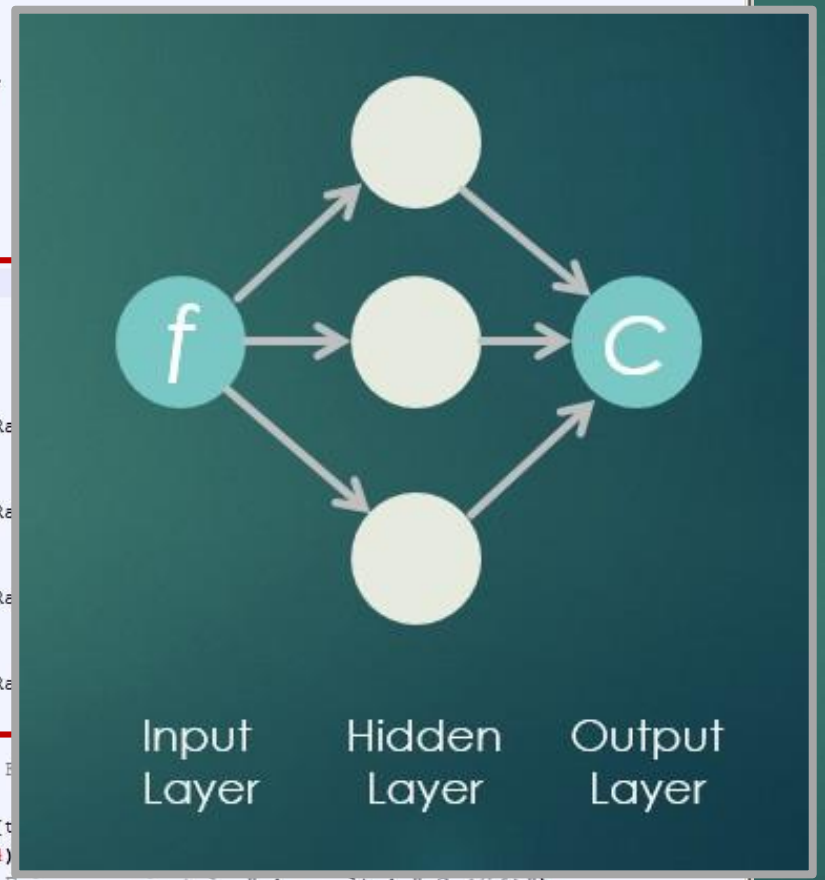
70 F / 21 C

99 F / 38 C

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRa

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRa

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRa

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRa
}
// run the network
var temp = prompt("Enter Degrees
var tempF = "." + temp;
var result = myNetwork.activate([t
result = String(result).slice(2,4)
document.write("<h1>" + temp + "  is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS



3. Train the Network on Data

a) Forward Propagation – activate the network; send data into the network from the input layer

b) Produce different outputs and compare the actual output to the intended output. The "weight" of the connections to the hidden layer influence the value of the output.

c) Back Propagation – calculate the difference between actual and intended result. Use this figure to adjust the weights of the connections to the hidden layer.

d) Do this **80,000** times! "Tune" the weights to produce better results.

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

4. Run or Test the Network

a) Use `prompt()` to create a dialogue box where you can enter degrees F

b) Activate the network by sending this number into the network. Produce an output and format the number.

c) Use `document.write()` to display the result (the output).

temp_machine_learning.html

```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

A number that was part of the training data

A number that was **NOT** part of the training data

temp_machine_learning.html

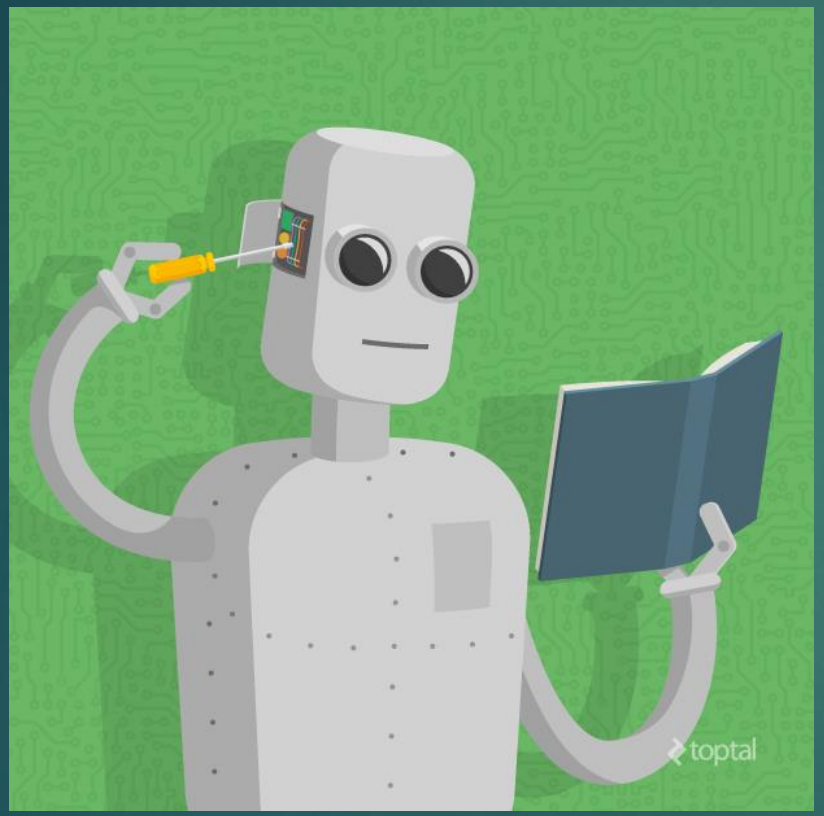
```
<html>
<head>
    <script src="http://gunkelweb.com/coms493/synaptic.js"></script>
</head>
<body>
<script>
//make the network
const { Layer, Network } = window.synaptic;

var inputLayer = new Layer(1);
var hiddenLayer = new Layer(3);
var outputLayer = new Layer(1);

inputLayer.project(hiddenLayer);
hiddenLayer.project(outputLayer);

var myNetwork = new Network({
    input: inputLayer,
    hidden: [hiddenLayer],
    output: outputLayer });

// train the network
var learningRate = .3;
for (var i = 0; i < 80000; i++)
{
    myNetwork.activate([0.30]);
    myNetwork.propagate(learningRate, [0]);

    myNetwork.activate([0.50]);
    myNetwork.propagate(learningRate, [0.10]);

    myNetwork.activate([0.70]);
    myNetwork.propagate(learningRate, [0.21]);

    myNetwork.activate([.99]);
    myNetwork.propagate(learningRate, [0.38]);
}
// run the network
var temp = prompt("Enter Degrees Fahrenheit");
var tempF = "." + temp;
var result = myNetwork.activate([tempF]);
result = String(result).slice(2,4);
document.write("<h1>" + temp + " F is approximately " + result + " C </h1>");
</script>
</body>
</html>
```

Hyper Te | length : 1124 lines : 46 | Ln : 22 Col : 1 Sel : 0 | 0 | Dos\Windows | ANSI as UTF-8 | INS

Improvements/Mods

- Increase the size of the data that is used for training
- Adjust the number of times this data is feed through the network.
- Adjust the way the network alters the weights; this is called the Learning Rate
- Increase the size of the network. Add more neurons to the hidden layer or add more layers. When there are numerous hidden layers, this is called “deep learning.”

v2 – Machine Learning



Summary

- Set-up a neural network (i.e. 1-3-1 network) and feed it some data
- Programmer does not know all the conversion data. The network "discovers" or "learns" conversion patterns from data.
- Variables effecting outcome: Quality and quantity of the data; number of layers in the network; learning rate (weight adjustment)

Summary

Algorithm

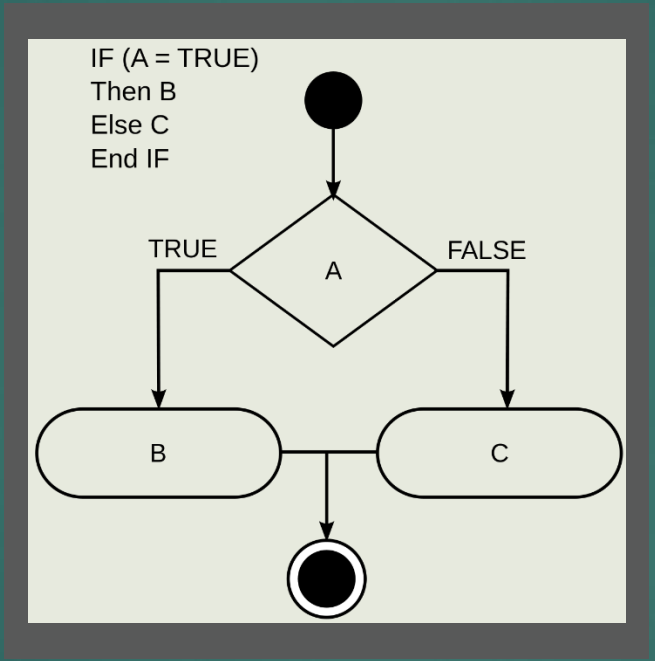
Input → → **Output**

Magic

Fear

Summary

GOFAI Algorithm

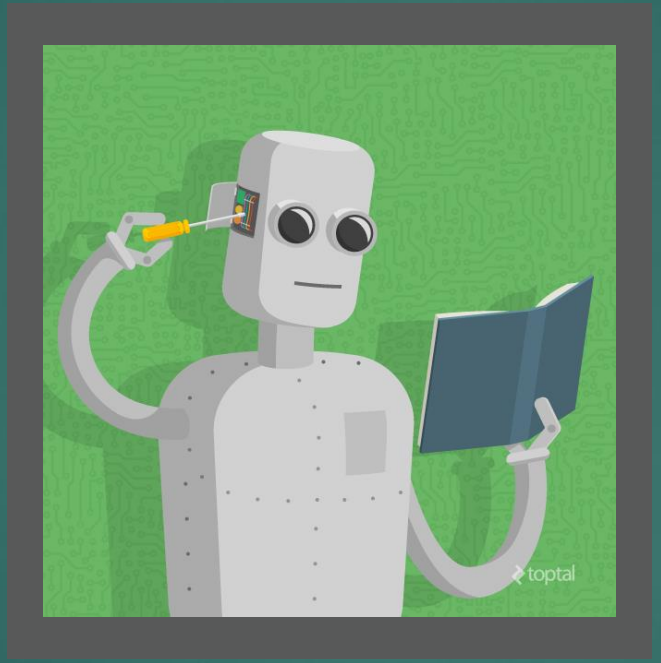
Input → 

→ **Output**

- Explicit step-by-step instructions
- Programmer can locate problems in the code and make edits to affect the output
- Proprietary Algorithms vs. Open Access

Summary

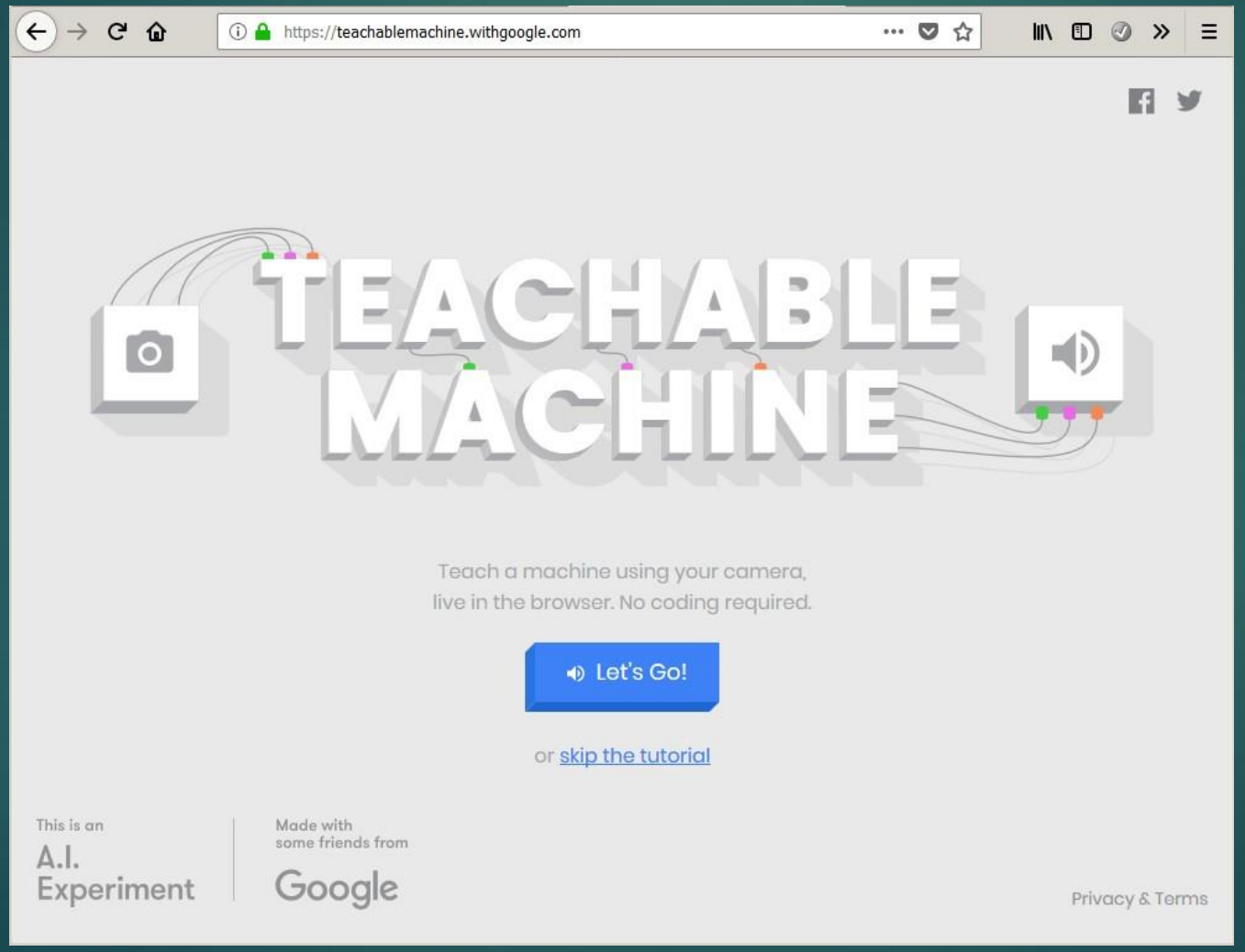
ML Algorithm

Input →  → **Output**

- Algorithm discovers patterns in data
- Potential problems – Error or bias in the training data
- Cannot inspect code to find the source of the error
- Programmer does not know what the program will do until it does it.

Learning More

https://teachablemachine.withgoogle.com/



Preview

- NLP & Computational Creativity
  - Intro to Communication & AI - ch. 5-6
  - Amper & Taryn Southern (videos)
  - Sunspring (video)