Large Language Models for Social Science

Foundations and Applications

Ali Amini

Center for Data Science, American University

January 7, 2026

The AI Bubble

- The United States produces **26% of global GDP**
- Yet U.S. stocks represent **over half of global equity markets**
- **Eight AI-oriented firms now account for 40% of the entire U.S. stock market**

Buffett Indicator (Market Cap / GDP)

- 70% $\rightarrow$ historically undervalued
- 150% $\rightarrow$ caution zone
- **220% today $\rightarrow$ historically extreme**

This reflects:

- Massive capital flowing into AI
- Winner-take-all tech economics
- Speculative expectations about future productivity

There Is No Education Bubble

- We do not know whether markets will crash
- But we do know that AI infrastructure is real and easy to learn

What Does GPT Stand For?

- **G — Generative**: Produces new text
- **P — Pre-trained**: Learned from massive data before you ever use it
- **T — Transformer**: A specific **neural architecture**

GPT is not a chatbot.
It is a:

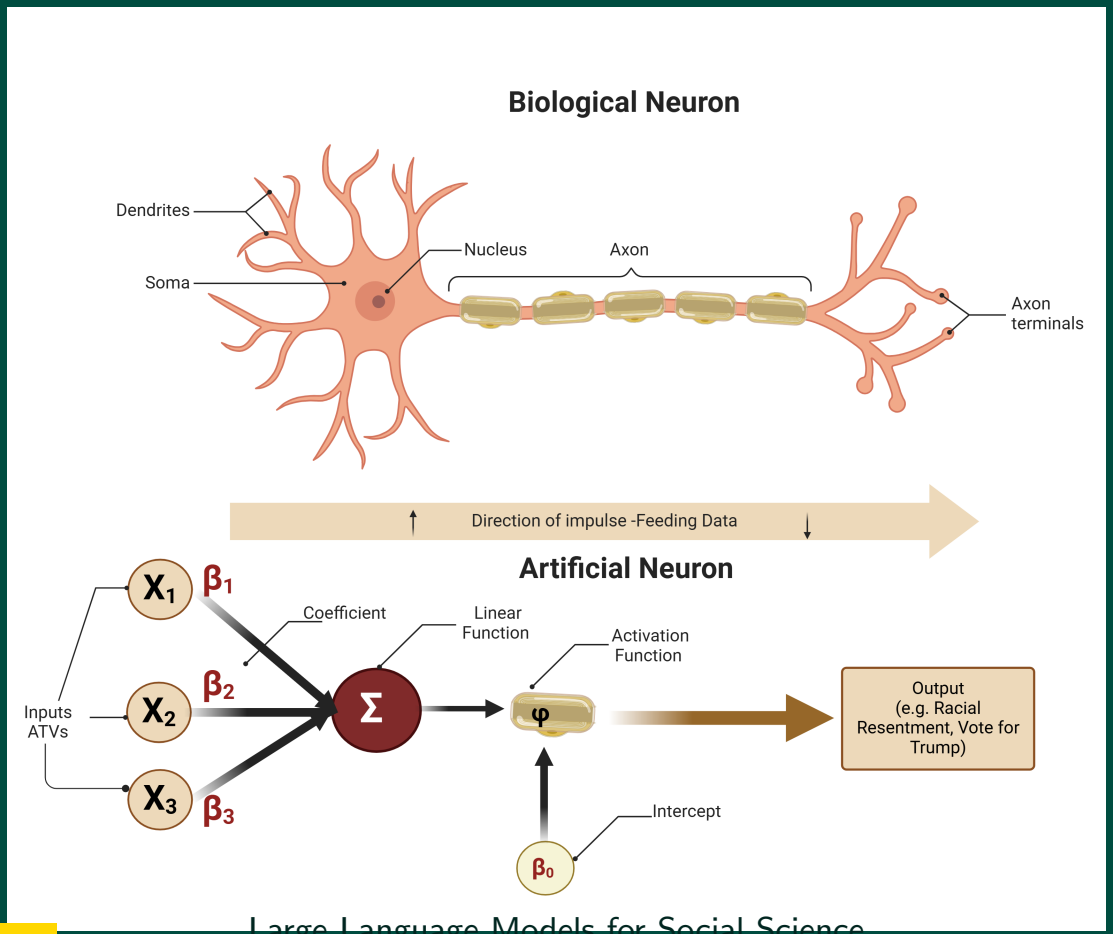
$$P(\text{next word} \mid \text{previous words})$$

trained on trillions of tokens.

Neural Architecture vs. Linear Model

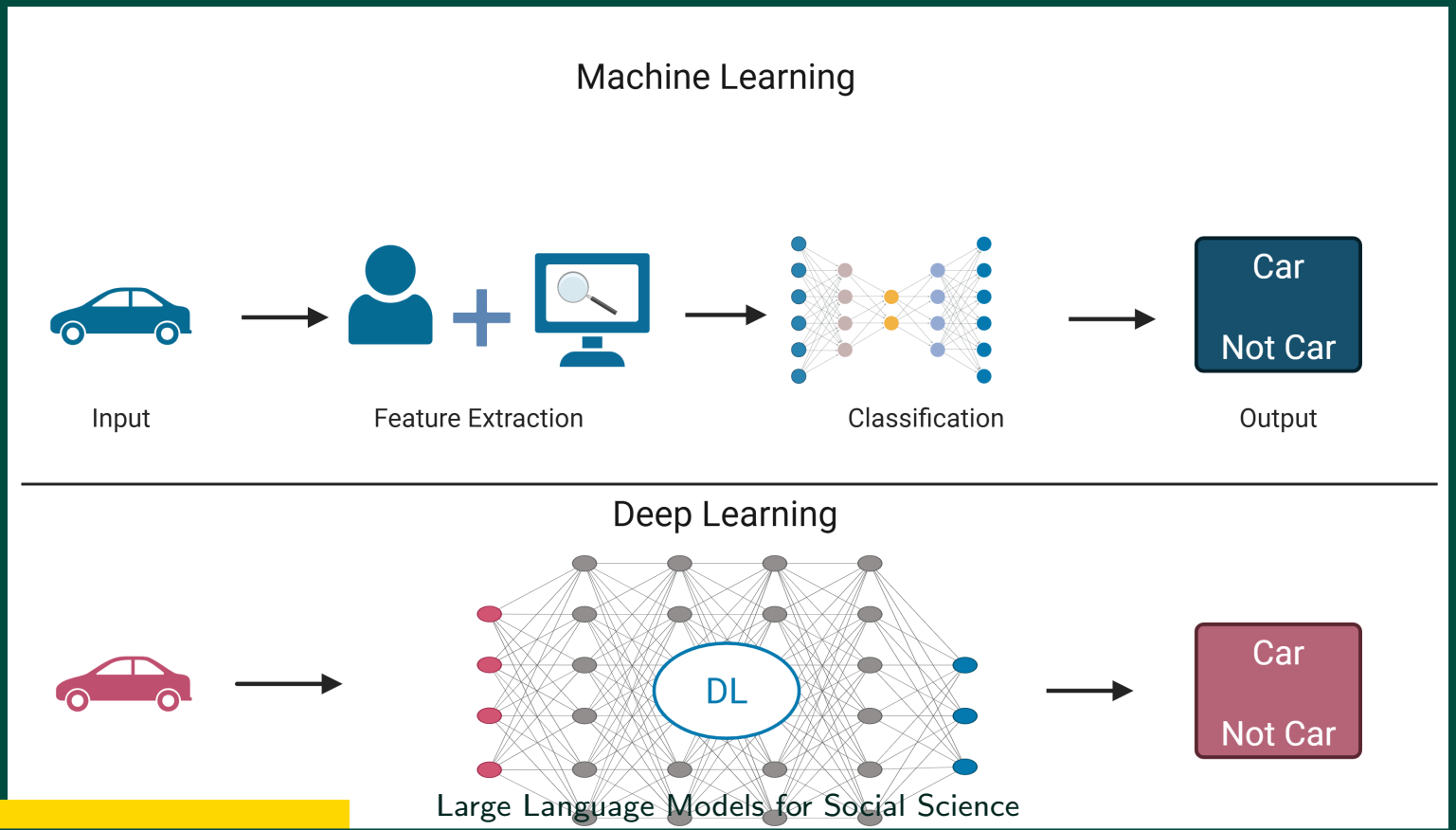
- **Linear models** combine inputs with fixed coefficients
- **Neural units** transform it through a learned non-linear activation

$$Y = \phi(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3)$$



Machine Learning vs Deep Learning

- **Classical machine learning** uses human-designed features + statistical models
  - Logistic regression, Random forests, Support vector machines
- **Deep learning** uses neural networks to learn features and predictions jointly
  - Multilayer neural networks, CNNs, **Transformers**



Random Forest vs Neural Network

Feature	Random Forest	Neural Network
Building Block	Decision trees using yes/no rules	Neurons passing weighted signals through layers
Logic	Simple conditional rules (e.g., "if income $>$ 50k")	Continuous transformations (linear algebra + calculus)
Feature	Human-designed inputs (e.g., income, race, education)	Automatically learned from raw data
Hardware	Efficient on CPUs	Often needs GPUs for large-scale training
Interpretability	Easy to inspect and explain	Often a "black box"

Bottom line: Random forests use rules; neural networks use learned representations.

What Is a Large Language Model?

A Large Language Model is a probabilistic model of language

$$P(\text{next token} \mid \text{previous tokens})$$

LLMs are trained on massive corpora of human text and learn:

- Grammar (how sentences are formed)
- Facts (what tends to be said about the world)
- Logic (how arguments are constructed)
- Ideology (how groups talk about politics, race, power)

They do not contain explicit rules.
They learn **statistical regularities** in how language is used across society.

What Do LLMs Learn About Society?

Because LLMs are trained on political and social text, they internalize patterns such as:

- "Lower taxes" $\rightarrow$ "Republicans"
- "Systemic racism" $\rightarrow$ "Democrats"
- "Immigration" $\rightarrow$ "threat" or "human rights"
- "Police" $\rightarrow$ "law and order" or "racial bias"

These are not programmed.
They emerge because these words co-occur millions of times in real political discourse.

LLMs become probabilistic maps of ideology, culture, and power.

What Is a Transformer?

A **Transformer** is the neural architecture that powers all modern LLMs (GPT, BERT, LLaMA, Claude).

- Processes all words in a sentence at once (not sequentially)
- Uses **self-attention** to decide what matters
- Builds layered representations of meaning

Instead of reading left to right, transformers look at the **entire context** and ask:

Which words matter for understanding this word?

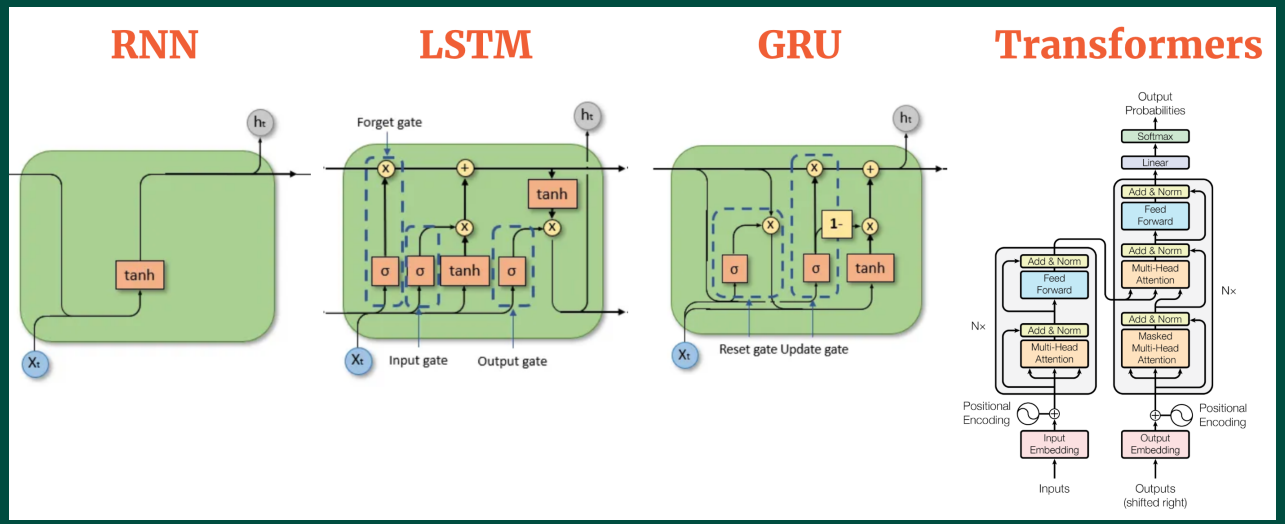
Attention is all you need

A Vaswani, N Shazeer, N Parmar... - Advances in neural ..., 2017 - proceedings.neurips.cc

... to attend to **all** positions in the decoder up to and including that position. **We need** to prevent ... **We** implement this inside of scaled dot-product **attention** by masking out (setting to $-\infty$) ...

☆ Save 99 Cite Cited by 211006 Related articles All 70 versions »

From Sequential to Parallel: The Evolution, (Revolution!)



The progression: RNN processes one word at a time → LSTM/GRU add memory gates → **Transformers process all words simultaneously**.

Why Transformers are a game changer:

- **Parallelization:** processes entire sentences at once, not word-by-word
- **Example:** "The bank by the river" — Transformer instantly sees all words and knows "bank" means riverbank. RNN/LSTM must wait to see "river" first.

Background: Sequential Models

Before Transformers, models processed text word-by-word sequentially.

RNN (Recurrent Neural Network)

- Processes left to right, passing hidden state forward
- *Problem:* forgets early words in long sentences

LSTM (Long Short-Term Memory) & GRU (Gated Recurrent Unit)

- Add memory gates to remember important information longer
- *Problem:* still sequential — can't process all words simultaneously

Key limitation: Sequential processing is slow and loses long-range context.

Attention Is All You Need

In 2017, Vaswani et al. showed that **attention alone** could replace sequential models.

The Transformer: processes all words in parallel using attention.

The core equation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V$$

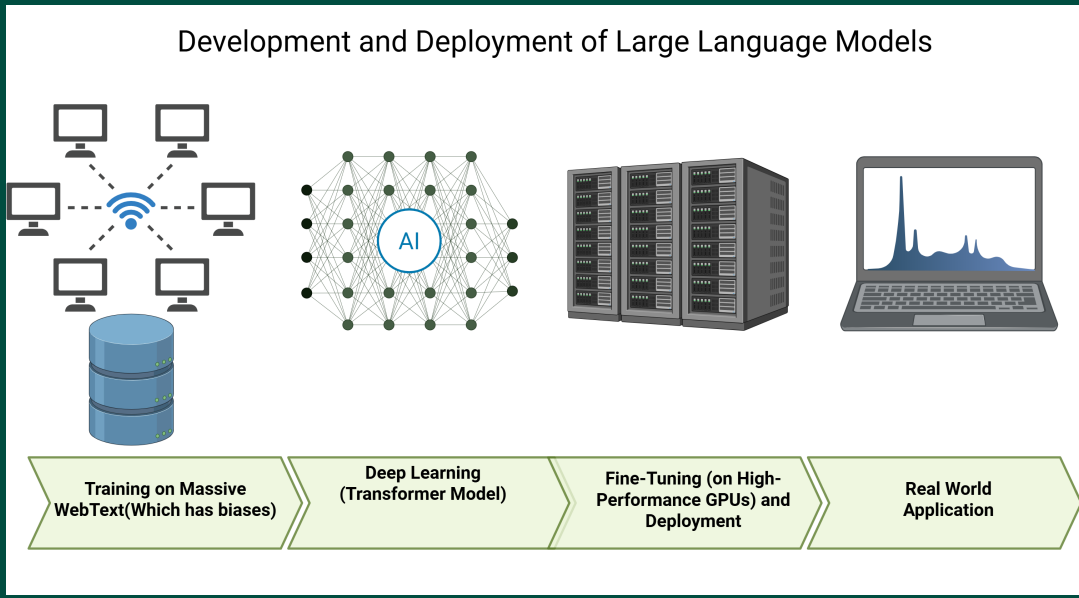
- **Q** (Query): what am I looking for?
  Example: "river" asks: *what words describe me?*
- **K** (Key): what do other words contain?
  Example: "wide", "deep", "flows" advertise their meanings
- **V** (Value): what information should I take?
  Example: "wide" provides its embedding/semantic content

Result: faster training and better long-range understanding.

From Transformers to Large Language Models

LLMs are transformers trained on massive text to predict the next word.

- **Tokenize:** Break text into pieces
  "Hello world" → [`Hello`, `world`]
- **Embed:** Convert to numbers
  `Hello` → [0.2, 0.5, 0.1, ...]
- **Transform:** Pass through attention layers
  Learn relationships between words
- **Predict:** Output next word probability
  "Hello __" → 85% *world*, 10% *there*, ...

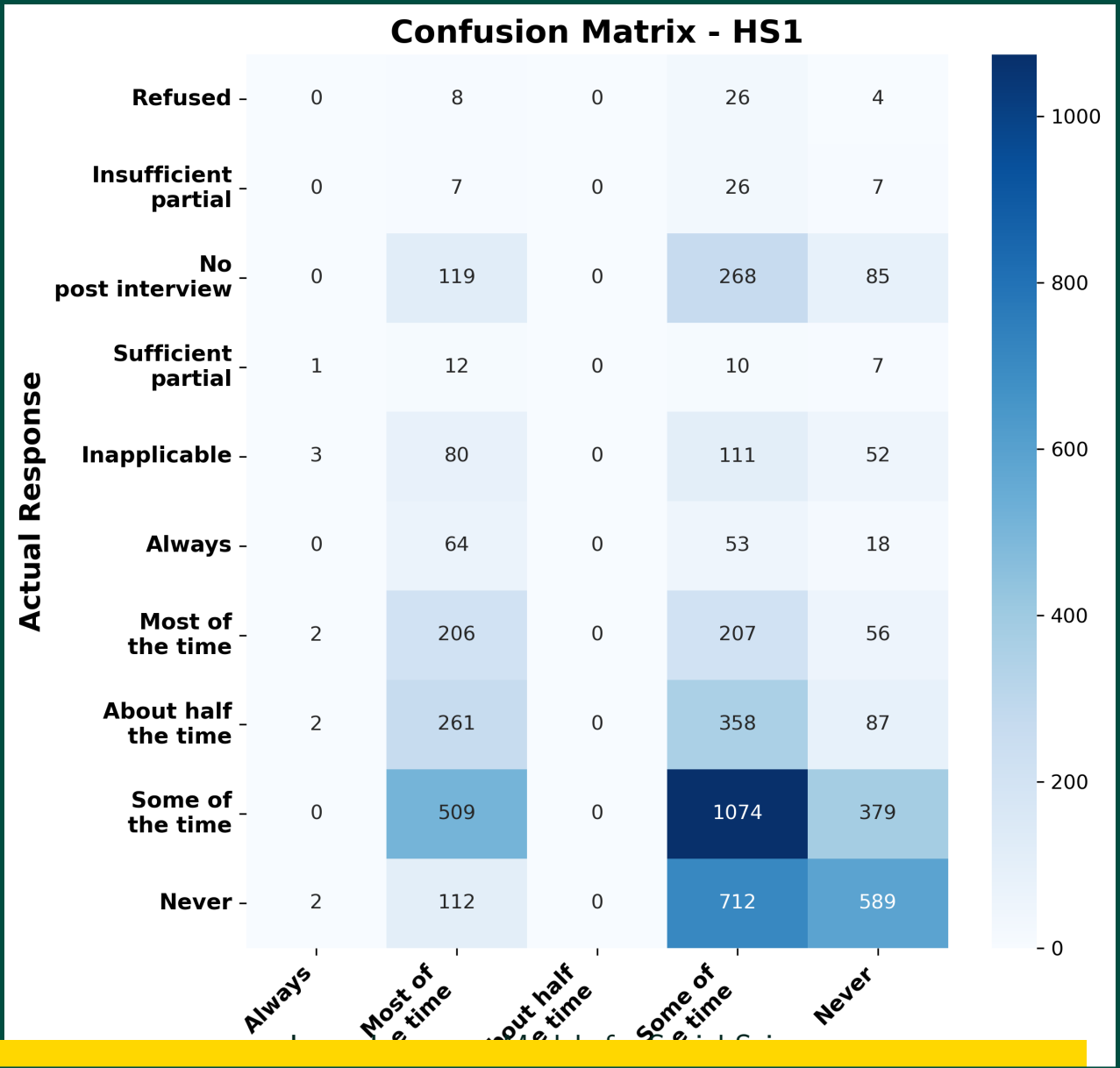


Application of LLMs: Racial Resentment and Hostile Sexism

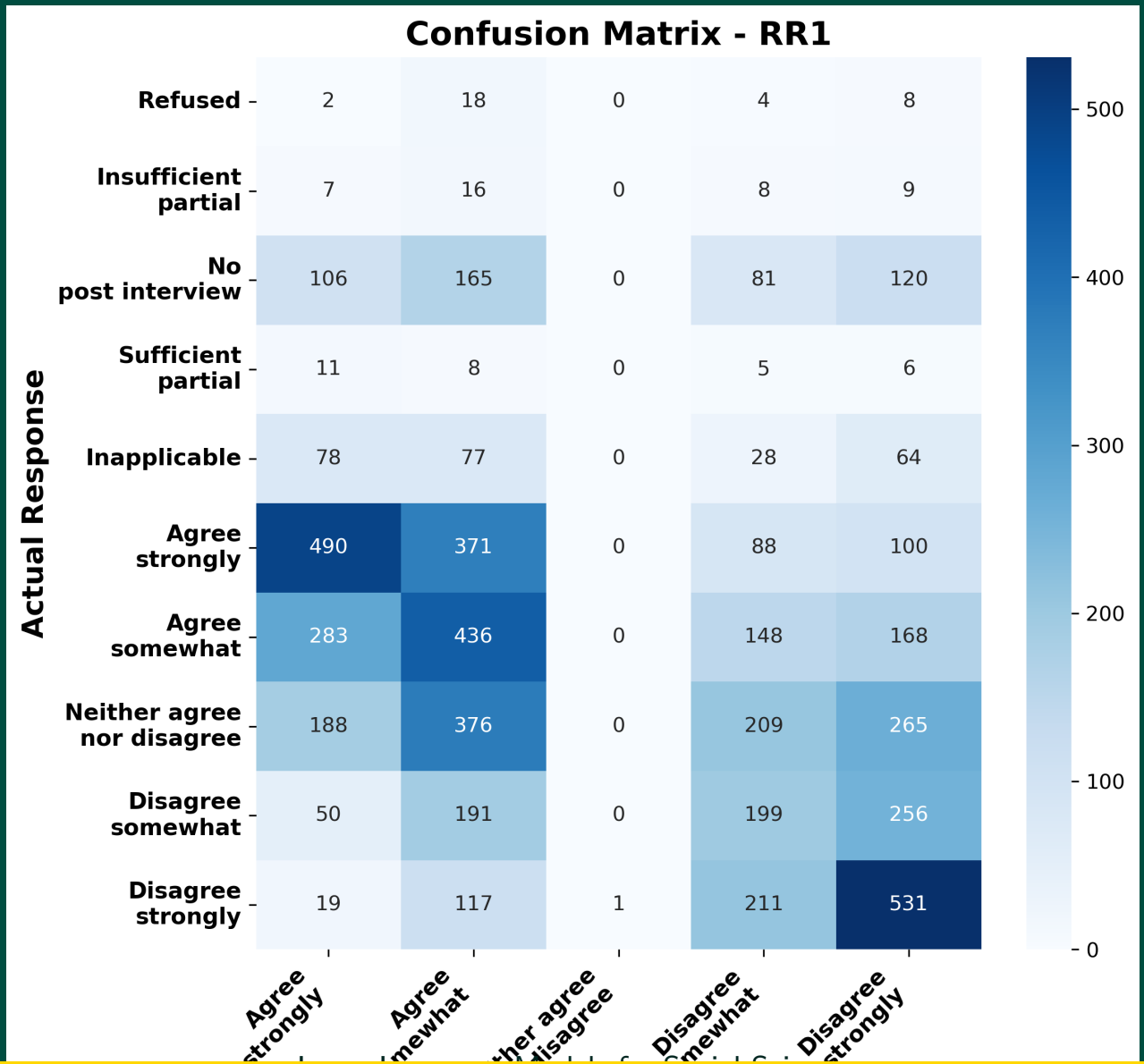
- **Hostile Sexism (HS1):** When women complain about discrimination, how often do they cause more problems than they solve?
- **Racial Resentment RR1:** Irish, Italian, Jewish and many other minorities overcame prejudice and worked their way up. Blacks should do the same without any special favors.

Response scale: Always – Never

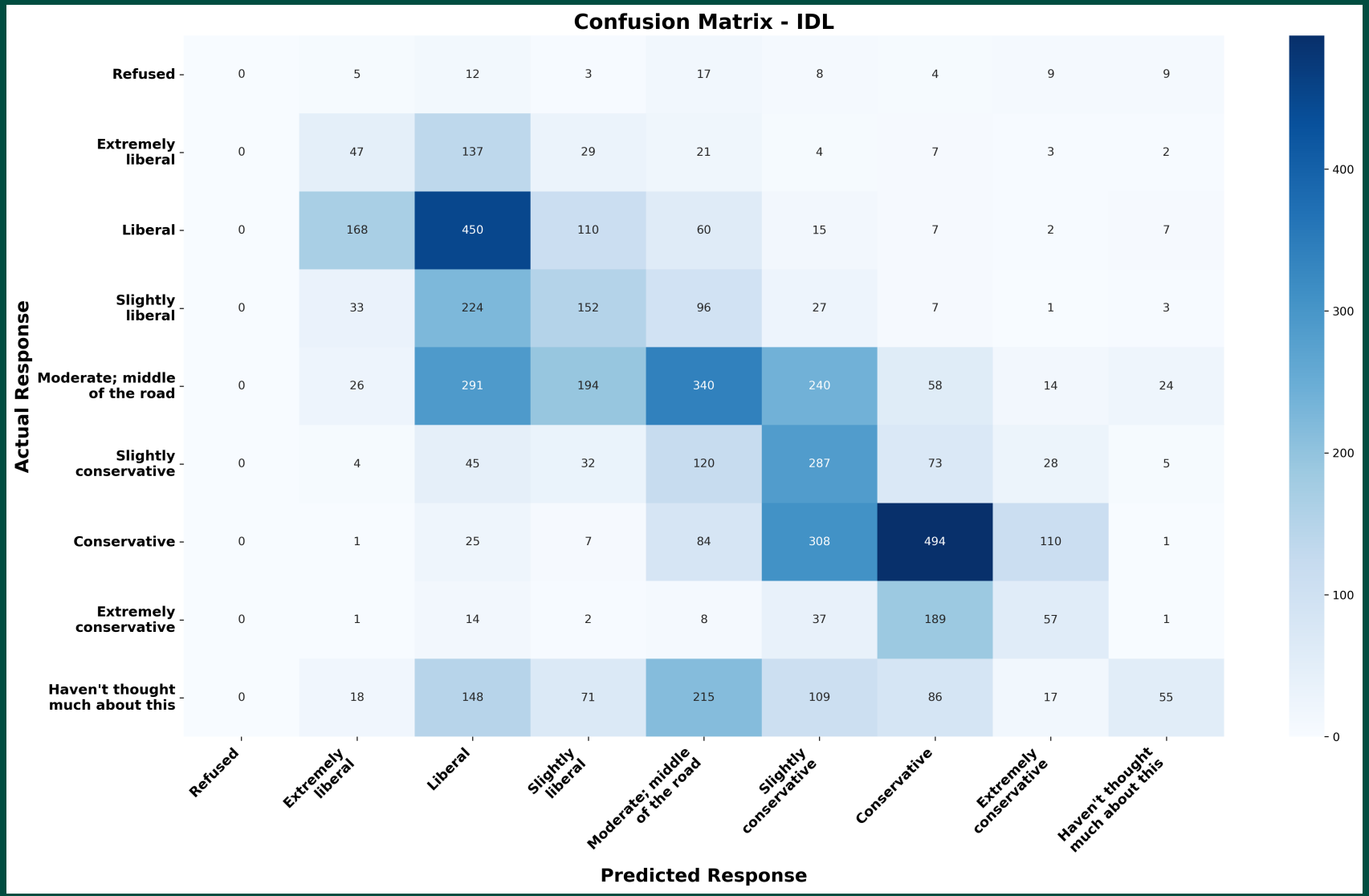
Hostile Sexism



Racial Resentment



Ideology


Confusion Matrix - IDL
Actual Response
Predicted Response
Refused
Extremely liberal
Liberal
Slightly liberal
Moderate; middle of the road
Slightly conservative
Conservative
Extremely conservative
Haven't thought much about this
0 5 12 3 17 8 4 9 9
0 47 137 29 21 4 7 3 2
0 168 450 110 60 15 7 2 7
0 33 224 152 96 27 7 1 3
0 26 291 194 340 240 58 14 24
0 4 45 32 120 287 73 28 5
0 1 25 7 84 308 494 110 1
0 1 14 2 8 37 189 57 1
0 18 148 71 215 109 86 17 55
400
300
200
100
0

Conclusion: What You Should Take Away

- **AI is infrastructure, not a fad:** market hype may fluctuate, but transformer-based systems are now embedded in research and industry.
- **LLMs are probability models of language:** they learn statistical regularities in text, including social and ideological structure.
- **Transformers scale because of attention:** self-attention enables context-sensitive representations without recurrence.
- **For social science, the value is operational:**
  - Measurement at scale (coding, classification, latent constructs)
  - Text-as-data pipelines (annotation, extraction, summarization)
  - New tools for prediction and validation (with careful evaluation)
- **LLMs systematically miss the political middle:** because they learn from polarized and expressive text, they overrepresent extreme and elite voices and underrepresent moderate, ambivalent, or weakly held opinions.

Thank You!

aliamini.org

Questions? Comments?