SESSION 4 REFERENCE GUIDE

Session 4: Two Models, One Space — Instructor Guide

AI + Research Level 2 — Supplementary Material

Concept: MODEL EVALUATION

Space: Sentiment Showdown
Models: 3 sentiment models with different training data, compared side by side

---

Time Breakdown (2 hours)

0:00–0:05 — Show-and-Tell

- Ask: "Did anyone try the between-session challenge from last time?"
- Quick share of anything students experimented with.

0:05–0:15 — The Hook: Watch Models Disagree

- Open the finished Sentiment Showdown Space.
- Type: **"The service was slow but the food was amazing."**
- Watch the three models give different answers.
- Ask: "Which model is right?" (Trick question — they all are, from their own perspective.)
- Try: **"lol this is SO bad it's actually good 😂"** — sarcasm breaks things.

0:15–0:50 — Live Build (35 minutes)

This is the most complex build so far — three models in one Space. Budget extra time.

1. **Create new Space on Hugging Face** — "Sentiment Showdown"
2. **Write requirements.txt** — `transformers`, `torch`, `gradio`
3. **Build app.py step by step:**
   - Import libraries
   - Load first model, test it alone

- Add second model, test both
- Add third model, write the comparison function
- Wire up Gradio with 3 output boxes
- Add examples

4. **Deploy and test**

Pacing note: If running long, skip the third model during live build and add it after testing the first two. Students can see the "add another model" pattern.

0:50–1:10 — Test and Explore (20 minutes)

Students suggest inputs. Chase disagreements.

Key teaching moments:
- "High confidence doesn't mean correct." A model can be 98% sure and still wrong.
- Sarcasm is hard for all three models.
- Mixed-sentiment text ("slow but amazing") forces models to pick a side.

1:10–1:25 — Spam Detector Thought Experiment (15 minutes)

Read this scenario to students:

> *Imagine you built a spam detector for email. You test it on 1,000 emails. 950 are real mail, 50 are spam. Your model predicts "NOT SPAM" for every single email — all 1,000.*
>
> ***Question 1:** What's the accuracy? (Answer: 95% — it got 950 out of 1,000 correct!)*
>
> ***Question 2:** Is this a good spam detector? (Answer: No! It never catches any spam.)*
>
> ***Question 3:** Would you ship this? Why not?*

Discussion points:
- Accuracy alone can be misleading.
- What matters depends on the task: a missed spam email is annoying, but a missed fraud alert is dangerous.
- **False positives** (real email marked as spam) vs. **false negatives** (spam that gets through) — which is worse? Depends on context.
- "How do we decide if a model is actually good?" → This is MODEL EVALUATION.

1:25–1:40 — Read the Model Cards (15 minutes)

Pull up model cards on Hugging Face for all three models. Point out:

- **Training data size and source** — this is WHY they disagree.
- **Intended use** — each model was built for a specific domain.
- **Limitations** — the model creators already know what fails.

Model Card URLs:
- Movie Review Model: https://huggingface.co/distilbert-base-uncased-finetuned-sst-2-english
- Twitter Model: https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest
- Product Review Model: https://huggingface.co/nlptown/bert-base-multilingual-uncased-sentiment

Key takeaway: "The Twitter model was trained on 124 million tweets. The default model was trained on movie reviews. That's WHY they disagree — they learned language from different worlds."

1:40–1:50 — Name the Concept: MODEL EVALUATION

- We've been doing model evaluation this whole session.
- Evaluation isn't just "is it right?" — it's "right for what? For whom? Measured how?"
- Different metrics for different tasks. Accuracy is just one number.

1:50–2:00 — Notebook Time

Share the Colab link in the Zoom chat.

Walk through together:
1. Run the setup cell
2. Load the three models one at a time — warn them "this takes a minute, be patient"
3. Run the first showdown comparison together
4. Show them the score table in the notebook — they'll fill it in during experiments

Say: "The notebook has all three models ready to go. Try the experiments — especially the sarcasm one. See if you can find the input that causes maximum disagreement."

Notebook skill being introduced: Installing packages with `!pip` (first cell), patience with long-running cells

GitHub skill being introduced: "Upload this notebook to your `my-ai-portfolio` repo — same as last time."

2:00 — Wrap Up

- Share the between-session challenge (see BETWEEN-SESSION.md).

Pre-Prepared Inputs That Cause Disagreement

Use these to demonstrate. Each one triggers interesting model behavior:

Input	What Happens
"The service was slow but the food was amazing."	Mixed sentiment — models pick different sides
"lol this is SO bad it's actually good 😂"	Sarcasm + emoji — confuses most models
"The movie was fine. Nothing special but not bad either."	Neutral/ambiguous — models handle neutrality differently (movie model has no neutral label!)
"I can't believe how terrible this is. Just kidding, it's great!"	Negation + reversal — tests understanding of context
"The product arrived on time and works as described."	Factual/neutral — but product model may read it as positive
"meh whatever i guess its ok lol"	Casual/informal — Twitter model handles this better
"This establishment has consistently failed to meet even the most basic standards of customer service."	Formal negative — all should agree, but confidence levels differ
"10/10 would not recommend"	Internet sarcasm — "10/10" looks positive, "would not recommend" is negative

---

Technical Notes

- **Memory:** Three models total ~1.5–2GB. This should fit on free HF CPU Spaces (16GB RAM), but models are loaded sequentially at startup to avoid memory spikes.
- **Speed:** First run after deploy is slow (downloading models). Subsequent runs are faster.

- **Token limit:** Input is truncated to 512 characters to stay within model limits.
- **If memory issues occur:** Move model loading inside the function so only one model is in memory at a time (slower but safer). Or replace the largest model with a smaller one.

Concept Connections

- **Session 1–3:** Students learned INPUT → MODEL → OUTPUT with one model at a time.
- **Session 4 (this session):** Same input, multiple models — now we need to evaluate which one is better.
- **Session 6 (upcoming):** Same models, different input domains — domain shift.