Using Generative AI for Social Media Analysis

IJC Research Methods Bootcamp, February 3, 2025

Dani Madrid-Morales

Set up

For this session you will need…

1) Access to ChatGPT (or Gemini or Claude).
2) Sample social media content for the hands on activities (available through the QR code on the right).
3) Activity Handout.
4) This PPT presentation.

Learning Objectives

By the end of the session, you should be able to:

1. Use **Generative AI** to analyse text from social media.
2. Apply different **prompt engineering techniques** (zero-shot, one-shot, few-shot) for better AI responses.
3. Evaluate and **validate** AI-generated outputs for accuracy.
4. Understand the **limitations** and **challenges** of using AI for social media analysis.

Overview

Why Use Generative AI for Social Media Analysis?

- Social media generates vast amounts of **unstructured data**.
  - GenAI enables rapid analysis of high-volume social media text and images.
- Most traditional NLP methods require **manual** feature engineering and/or labeled datasets.
  - GenAI reduces reliance on manual annotation and rule-based methods.
- GenAI enables **automated insights** through summarisation, classification, and content generation.
  - GenAI has enhanced ability to detect emerging patterns.

Comparing Approaches

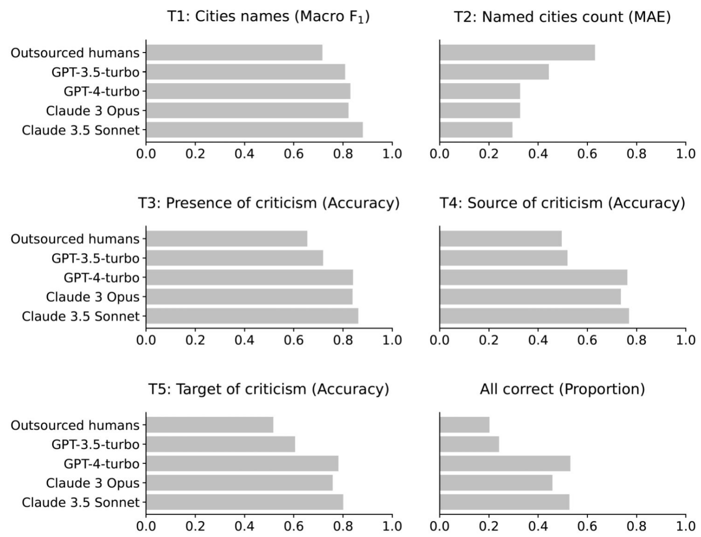
Approach	**Strengths**	**Weaknesses**
Manual Content Analysis	High construct validity, widely accepted by research community	Expensive, impossible to apply at scale
Keyword-Based Analysis	Easy to implement, fast	Lacks context, high false positives
Statistical NLP Approaches	Well-established, interpretable	Requires large annotated datasets, struggles with sarcasm & slang
Machine Learning Models	Higher accuracy, adaptable	Computationally intensive, requires labeled training data
Generative AI (e.g., Gemini)	Context-aware, handles multimodal data	Can generate hallucinated content; lack of transparency.

Advantages & Disadvantages of Generative AI

- **Advantages**
  - Adaptability across languages and styles.
  - Versatile: Can summarise, categorise, and fact-check.
  - Scalable: Handles large datasets efficiently.
  - Interactive: Users can refine outputs iteratively.
- **Disadvantages**
  - Lack of Transparency: Process can be a “black box”
  - Not great for Open Science: Can be hard to replicate results.
  - Bias and Hallucinations: AI may generate misleading responses.
  - Context Dependence: Struggles with sarcasm and cultural nuances.

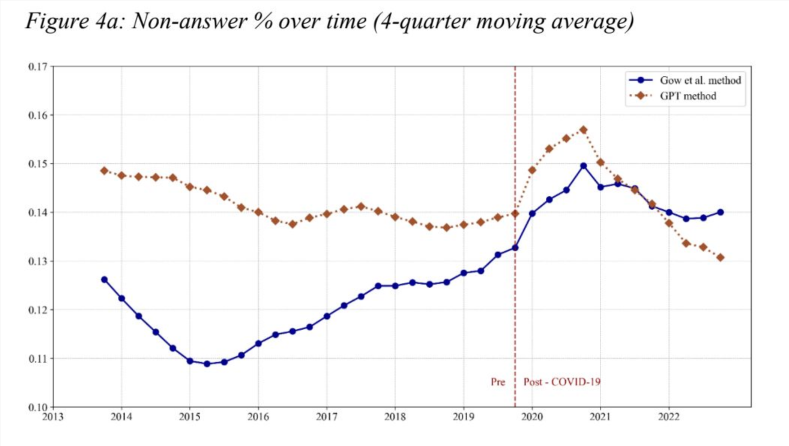
Example 1 | Bermejo et al. (2024)

Figure 1 Overall performance, across tasks and coding strategies



- **Data**: 210 news articles on fiscal consolidation, requiring political and economic analysis.
- **Approach**: Compared LLMs to human coders on five text analysis tasks using a zero-shot learning approach.
- **Evaluation**: LLMs consistently outperformed outsourced human coders in accuracy, cost, and efficiency, suggesting they are a viable replacement for large-scale text analysis

Example 2 | de Kok (2024)



- **Data**: Earnings conference calls, specifically identifying "non-answers" given by executives.
- **Approach**: Used GPT-4 to detect evasive responses in earnings calls, improving upon previous machine learning methods.
- **Evaluation**: Achieved 96% accuracy and reduced error rate by 70% compared to prior approaches, demonstrating LLMs' effectiveness in automating complex text classification

Example 3 | Hohenwalde et al. (2025)

Prompt	Codebook			Optimized prompt			NERC pipeline		
Class	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Science	.35	.79	.49	.38	.75	.50	.91	.96	.94
Advocacy	.00	.00	.00	.32	.55	.41	.84	.70	.76
Politics	.35	.69	.47	.34	.55	.42	.92	.91	.91
Others	.05	.28	.08	.05	.27	.09	.57	.55	.56
Overall	**.19**	**.44**	**.26**	**.27**	**.53**	**.36**	**.81**	**.78**	**.79**

Table 2: Precision, Recall and F1-scores for different prompting strategies using gpt-3.5-turbo.

- **Data**: Analysed 2,883 German news articles covering four scientific topics.
- **Approach**: Tested different LLMs for categorising societal actors, using different prompting strategies and Named Entity Recognition (NER).
- **Evaluation**: Best-performing model achieved an F1-score of 0.82, showing promise but still struggling with nuanced actor categories.

What can GenAI do for us?

Sentiment & Stance Analysis: Identifies sentiment and stance dynamically, understanding irony, sarcasm, and implicit meaning.

Thematic Analysis: Maps evolving narratives, ideological framings, and shifts in media discourse across time.

Classification & Categorisation: Performs few-shot and zero-shot classification, adapting to new discursive categories without extensive labelled data.

Framing Analysis: Detects patterns in media framing, issue salience, and cross-platform agenda-setting.

What can GenAI do for us?

Topic Clustering: Identifies latent themes without predefined categories, adapting to new issues dynamically.

Network Analysis & Influence Mapping: Generates synthetic network maps, identifies key influencers dynamically, and simulates narrative spread.

Multimodal Analysis: Interprets memes, GIFs, and image-text relationships, detecting implicit messages and cultural references.

Image & Context Recognition: Identifies visual narratives, recurring symbols, and thematic representations.

University of Sheffield

Hands-on Activity

GenAI Workflow

1. Ask RQs	2. Strategise	3. Iterate	4. Validate	5. Scale up
Identify clear theoretical constructs that can be measured quantitatively.	Design effective prompts and determine the best way to structure the input data.	Test different prompts. Compare multiple responses to identify inconsistencies.	Cross-check AI-generated insights against human-coded data. Use multiple folds to assess reliability.	Apply (and automate) the refined workflow to larger datasets for broader analysis.

Prompt Engineering

Prompt engineering is the process of designing effective prompts to get the desired output from a conversational AI tool.

Prompt engineering techniques enhance the quality of Large Language Model (LLM) outputs by focusing on **clarity**, specificity, providing **context**, and using **iterative** refinement and **examples**.

Prompt Engineering | Best Practice

Clear and Specific Prompts: The more clear and specific the prompt is, the more likely it is to get a useful output.

Contextual Prompts: LLMs do not have prior knowledge, so prompts must provide context. A prompt that lacks context may not produce the desired output.

Using Verbs in Prompts: Beginning a prompt with a verb ("create," "summarize," "classify") can guide the LLM to produce useful output.

Iteration: Prompt engineering is an iterative process. Evaluating the output and revising the prompt is key to improving results. This involves identifying any issues such as missing context, and adjusting the prompt's wording.

Prompt Engineering | Approaches

Few-Shot Prompting: Using examples in prompts can help LLMs respond more effectively. This technique, known as few-shot prompting, involves providing two or more examples to clarify the desired format, phrasing, or pattern.

Types of prompting:

- Zero-shot: AI responds without prior examples.
- One-shot: AI is given one example before inference.
- Few-shot: AI is given multiple examples for better understanding.

Prompt Engineering | Approaches

- **Zero-shot prompting**: No prior examples, AI infers from knowledge. Less accurate in some tasks.
  - Example: 'Summarise the main themes in these tweets about climate change.'
- **One-shot prompting**: AI gets one example before inference.
  - Example: 'This post is misinformation. Classify the following posts.'
- **Few-shot prompting**: Multiple examples improve output consistency. Can make responses less flexible and creative.
  - Example: 'Here are three examples of engagement types. Classify these new posts.'

Critically Assessing GenAI Output

When evaluating the output of a Large Language Model (LLM), consider the following key questions:

- Is the response **accurate** and factually **correct**?
- Does the output reflect **fairness** and minimise **bias**?
- Is the information provided **comprehensive** and sufficiently detailed?
- Is the response aligned with the **objectives** of my project or task?
- Does the model generate **consistent answers** when given the same prompt multiple times?

Measuring Performance

- Compare AI results with a human-coded subsample to assess reliability.
    - You can use common metrics such as **Cohen's Kappa** or **Krippendorff's Alpha** to measure consistency between human annotators and AI outputs, or use metrics common in machine learning such a **F1 Score**, **Precision** or **Recall**.
- If the GenAI model isn't performing as well as expected, consider
    - **Error Analysis**: Identify recurring AI misclassifications to refine prompt strategies and improve model reliability.
    - **Cross-Validation**: Run AI models on multiple folds of data (subsets) to test generalisability and prevent overfitting.

University of Sheffield

Recap

Key Takeaways

- Use GenAI as an **augmentation tool**, not a replacement.
- Validate outputs through **human oversight** and **iterative refinement**.
- Recognise **potential biases** in AI-generated insights.
- Implement multi-step validation **workflows** (e.g., prompt variation, model comparison…).

GenAI Workflow

1. Ask RQs	2. Strategise	3. Iterate	4. Validate	5. Scale up
Identify clear theoretical constructs that can be measured quantitatively.	Design effective prompts and determine the best way to structure the input data.	Test different prompts. Compare multiple responses to identify inconsistencies.	Cross-check AI-generated insights against human-coded data. Use multiple folds to assess reliability.	Apply (and automate) the refined workflow to larger datasets for broader analysis.

Scaling up

- Using **APIs** enables batch processing of large datasets, improving efficiency and scalability (at a cost).
- **Automate** AI workflows into pipelines to handle data retrieval, processing, and analysis at scale.
- **Evaluate performance** across different AI models to enhance reliability and mitigate biases.
- Leverage **cloud-based solutions** to process high-volume data efficiently.
- Implement **parallel processing** for faster analysis of large datasets.

Get in touch: d.madrid-morales@sheffield.ac.uk

www.sheffield.ac.uk