LLM-Generated "**Synthetic Users**" for Qualitative Research — A Validation Protocol

Keerthana Govindarazan* (Ideation, Lit Review, Protocol Development, Coding)

Hui Min Lee* (Lit Review, Protocol Development, Coding)

Temilade Adeeko* (Literature Review)

*PhD Student, Penn State, USA

UX and market research teams are starting to use LLM “synthetic users” for interview studies to derive product-related insights

<u>But there is need to evaluate systematically:</u>

1. which LLM configuration settings affect interview response quality

2. how personas should be created

3. when LLM-generated interview data is useful vs misleading

Research Questions

RQ1 — Configuration & Response Quality

How do differences in LLM configuration settings (model, temperature, API vs consumer platform, zero-shot vs few-shot memory) and interview method (LLM-moderated interviewing vs human interviewing vs LLM-generated themes and quotes) affect the depth, coherence, and realism of synthetic interview responses?

RQ2 — Persona Design

How does the level and type of persona detail (explicit demographic traits vs context-rich backstory) influence contextual nuance, bias, and stereotyping in interview responses?

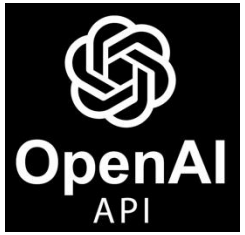
RQ3 — Research Use-Cases

Under what conditions are LLM-generated interview transcripts useful for exploratory or ideation-focused research—and when do they become misleading or unsuitable as substitutes for human participants?

Method (in-progress)

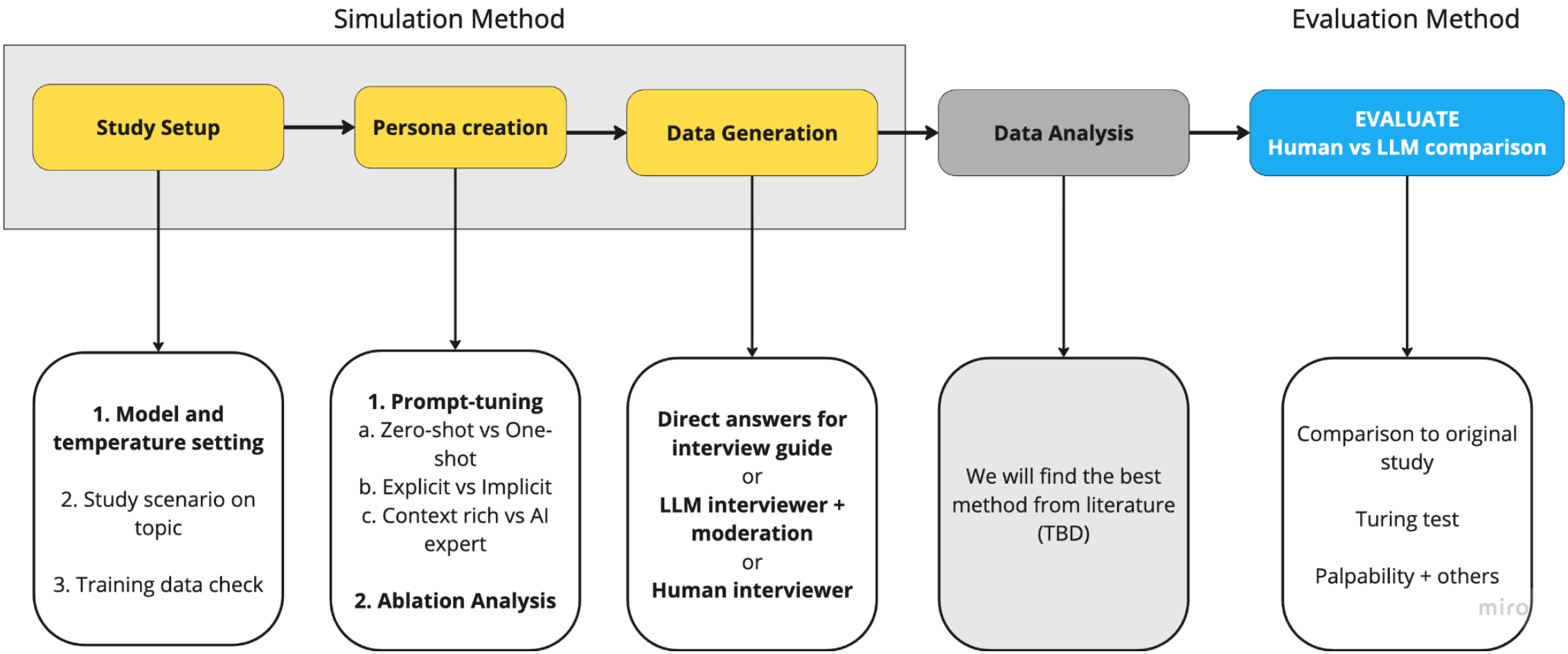
I replicate published HCI qualitative interview studies

Generate LLM interview transcripts by systematically varying LLM settings & persona conditions and benchmark against human interview results



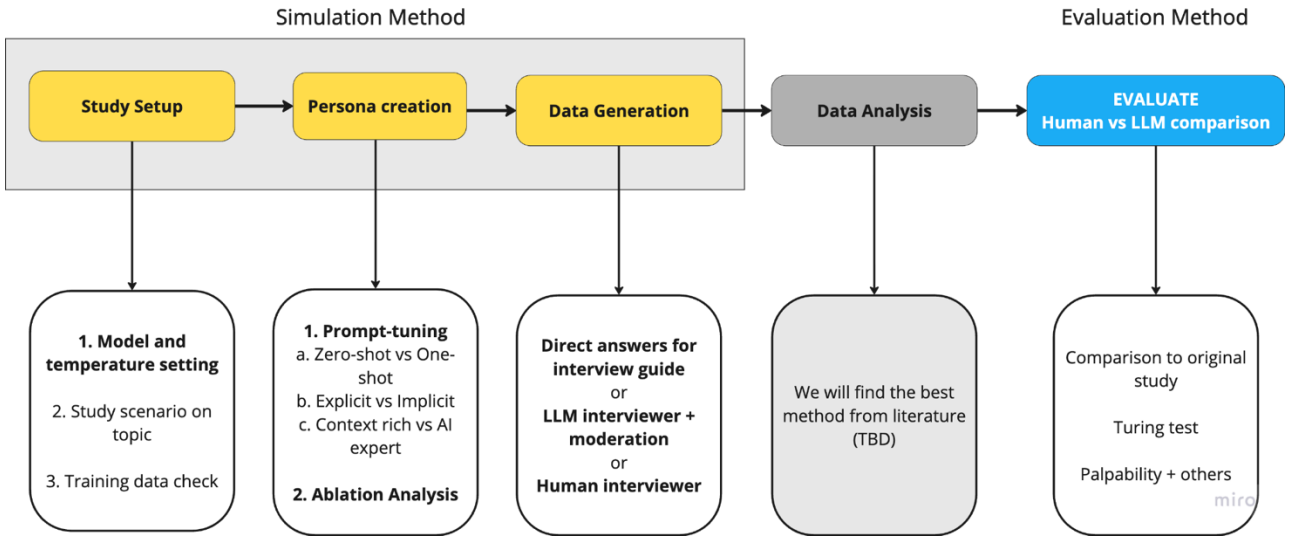
Goal: To understand how each configuration changes the quality of insights produced.

Protocol to be tested



Key LLM Configuration Factors

- **Zero-Shot vs Few-Shot vs Platform Memory**
  - whether transcript history shapes persona consistency
- **Explicit vs Implicit Personas**
  - demographic traits vs context-rich identity cues
- **Human Interviewer vs LLM Interviewer + Moderator**
  - probe behavior & response adaptation
- **LLM-as-Subject vs LLM-as-Expert**
  - simulate respondent vs summarize themes
- **API vs Consumer Platform**
  - accessibility trade-offs for non-technical researchers



UX Research Impact

This protocol will help teams:

- decide when LLM interview data is appropriate
- understand what different LLM setups are good for
  - ideation & early exploration
  - scenario prototyping
  - rapid hypothesis testing
- identify risk points:
  - stereotyping
  - loss of contextual depth
  - persona drift
  - saturation concerns and non-meaningful insights
- choose LLM configurations based on intended research purpose

```python
# ============================
# 1) CONFIG OBJECTS & TYPES
# ============================
PersonaStrategy = Literal["zero_shot_explicit","few_shot_explicit","implicit_context_rich"]
RoleMode = Literal["llm_as_subject","llm_as_expert"]
InterviewerMode = Literal["human_interviewer","llm_interviewer","llm_interviewer_with_moderation"]
FewShotMemoryMode = Literal["summary","full"]

@dataclass
class ModelConfig:
    model: str = "gpt-4o-mini"  # EDIT model
    temperature: float = 1.0    # EDIT temperature
    seed: Optional[int] = 42    # EDIT - helps reproducibility when supported

#study and method setup
@dataclass
class StudyConfig:
    project_name: str = "llm_interview_study1"             # EDIT project name
    today_str: str = datetime.now().strftime("%B %d, %Y")
    threshold: int = 80                                    # EDIT threshold to activate probe question
    max_probes_per_question: int = 1                       # EDIT number of probes
    persona_strategy: PersonaStrategy = "zero_shot_explicit" # EDIT study method
    role_mode: RoleMode = "llm_as_subject"                 # EDIT study method
    interviewer_mode: InterviewerMode = "llm_interviewer"  # EDIT study method
    few_shot_memory_mode: FewShotMemoryMode = "summary"    # EDIT study method. summary/full.
    out_dir: str = "./runs"                                # EDIT where the output is stored - filepat
    run_notes: str = "Pilot test with zero-shot persona"   # EDIT put in any notes about the run to be

#persona variables
@dataclass
class Persona:
    name: Optional[str]
    demographics: Dict[str, Any]
    implicit_context: Optional[str] = None

#interview topic
@dataclass
class InterviewGuide:
    topic: str
    questions: List[str]
    # note: probes are now DYNAMIC (no templates); keeping type for compatibility
    probes_library: Dict[str, List[str]] = None

# cost per response
# Optional: token->cost estimates (fill with your real prices per 1K tokens)
PRICE_USD_PER_1K = {
    "gpt-4o-mini": {"prompt": 0.00015, "completion": 0.00060},  # EDIT with actual pricing if desired
}
```

The End.

Contact: kmg6763@psu.edu
Portfolio: www.govindarazan.com