MaxDiff Analysis

To figure out which messages performed best I used an LLM to extract topics and rate each message based on how aggressive it was (just a scale from 1 to 5 with 5 being most aggressive). This made it easy to plot (Figure 1) the average MaxDiff scores by category.

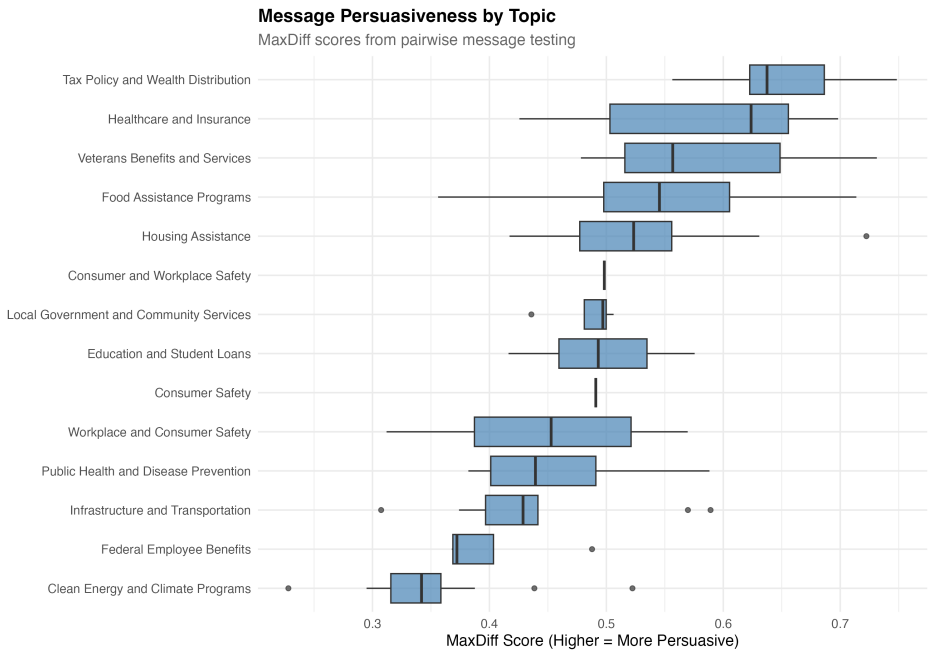

Figure 1: Message Performance by Topic

Overall, messages about tax cuts and wealth inequality performed best in the pairwise comparisons, followed by messages about healthcare, veterans benefits and services, and food assistance programs. Messages that performed poorly focused on climate change and clean energy, federal employees and infrastructure.

Figure 2 plots the average MaxDiff score by the LLM-coded aggressiveness of each message. Only values of 3 or 4 were assigned, but a first pass through the data suggests that more aggressive messages performed better. With more time, I would want to:

1. Ensure that the topic classification was accurately capturing the content of each message (not just policy but maybe broader political messages too). Or build out a predefined list of topics and categorize each message according to those.

2. Validate the LLM ratings of tone and aggression. Could do this by even asking respondents to rate the aggression of a message and using that as training data. Get more fine-grained classification of aggression (not ideal that the LLM only coded values of 3 and 4).

3. Do proper statistical inference on whether the differences in message performance between topic categories and aggression ratings were distinguishable from zero.

Figure 2: Message Performance by Tone

Housing Policy Survey

The housing policy survey reveals several seemingly supportive findings for pro-housing development policies:

- 54% say building new housing in their community would be **good** (vs. 46% bad)
- 49% know local government has most control over land use decisions
- 66% support requiring local governments to approve projects meeting state standards (net +51%)
- Only 27% oppose eliminating single-family-only zoning (net -24%)

However, I would be **extremely** cautious about using the results of this survey to say anything at all about the likely success of a campaign advocating for policies that allow local governments to approve housing projects. Making inferences about candidate / campaign performance based on polling that simply asks respondents to indicate their degree of support for policies is not a good idea.

First, policy support doesn't always translate to vote choice. Just look at states where referendums on enshrining abortion rights or raising the minimum wage passed at the same time that voters elected anti-choice and anti-high minimum wage candidates. Elections are choices between candidates that come with partisan labels and positions on other issues. That means that vote choice invariably involves making trade-offs.

There is even some evidence of this in the survey toplines. In question 60, 66% of respondents supported requiring local governments to approve housing plans when those plans met clear standards. But when the same question was asked with more context and respon-

dents were given different sides of the argument in question 68 that number dropped to 42%. Likewise, many "supportive" results have large "not sure" responses (23-36% on key questions), suggesting persuadable voters who could swing either way. We'd want to know how those persuadable voters move when targeted with messaging from the campaign on this issue.

The survey reveals that voters also prioritize many other issues, such as healthcare, inflation, education, crime, etc. Nothing in the survey tells us whether housing policy would move votes for undecided voters in a competitive race.[1]

I would say the survey reveals that housing policy is not inherently toxic. Support appears sensitive to framing, and combined with limited knowledge, we'd need to investigate how informing voters and treating them with campaign messaging moves vote choice. Which is the key problem with the survey: it tell us nothing about how these positions relate to vote choice. To make better informed inferences, we would need data on vote choice in head-to-head match-ups with party labels, issue priority rankings, and experimental tests of opposition messaging.

Question Bank Design

I designed a relational database with 10 tables to store 120+ survey questions, supporting diverse question types, reusability across surveys, and flexible querying.

Core Design Decisions

1. Separation of Questions and Response Options

Two-table design: `questions` stores question text and metadata; `response_options` stores predefined options only when needed (Likert scales, multiple choice). Open-ended and numeric scale questions require no options. This avoids fragile delimited strings and accommodates structural diversity.

2. Question Type Taxonomy

Eight question types (Likert, multiple choice, multiple select, open-ended, numeric scale, ranking, message test, grid) with validation flags (`requires_options`, `allows_multiple_selection`). Enables proper survey software rendering and analysis automation.

3. Dual Organization: Topics + Tags

Topics: Hierarchical taxonomy (Policy → Housing) for browsing. *Tags*: Many-to-many labels for cross-cutting concerns ("experimental," "core_demographic"). Provides both structure and flexibility since questions often span categories.

4. Conditional Logic

[1] On top of all of this, the question is unclear about what office the candidate is running for, which I would expect to matter a great deal (e.g. mayor of a small city vs President; this policy might play in the former but would be awful advice for a campaign in the latter).

`parent_question_id` and `display_condition` fields enable skip logic (e.g., "Who did you vote for?" shows only if prior response = "Yes"). Conditions stored as text for survey software flexibility.

5. Versioning and Reusability

Fields `version`, `is_active`, `times_used`, `last_used_date` support trend analysis. Updating questions creates new versions rather than deleting old ones (preserves historical comparability). Tracks usage to identify "greatest hits."

6. Survey Deployment Tracking

`surveys` and `survey_questions` tables link questions to fielded surveys, recording question ordering and field dates. Enables queries like "Show all housing questions fielded in 2024."

Implementation Results

`populate_question_bank.py` loaded 120 questions from CSV, inferring types and topics:

- 70 multiple choice, 29 Likert, 14 open-ended, 7 multiple select, 3 numeric, 2 ranking
- 495 response options
- Organized by 8 topics (Demographics, Vote Choice, Housing, etc.)

Example Queries

Find active housing questions:

```
SELECT * FROM v_questions_full
WHERE topic = 'Housing' AND is_active = TRUE;
```

Find frequently reused questions:

```
SELECT question_id, question_text, times_used
FROM questions WHERE times_used >= 5
ORDER BY times_used DESC;
```

Code: `src/create_question_bank.sql` (database schema), `src/populate_question_bank.py` (data loader)