Quality Diversity through AI Feedback - Supplementary Material

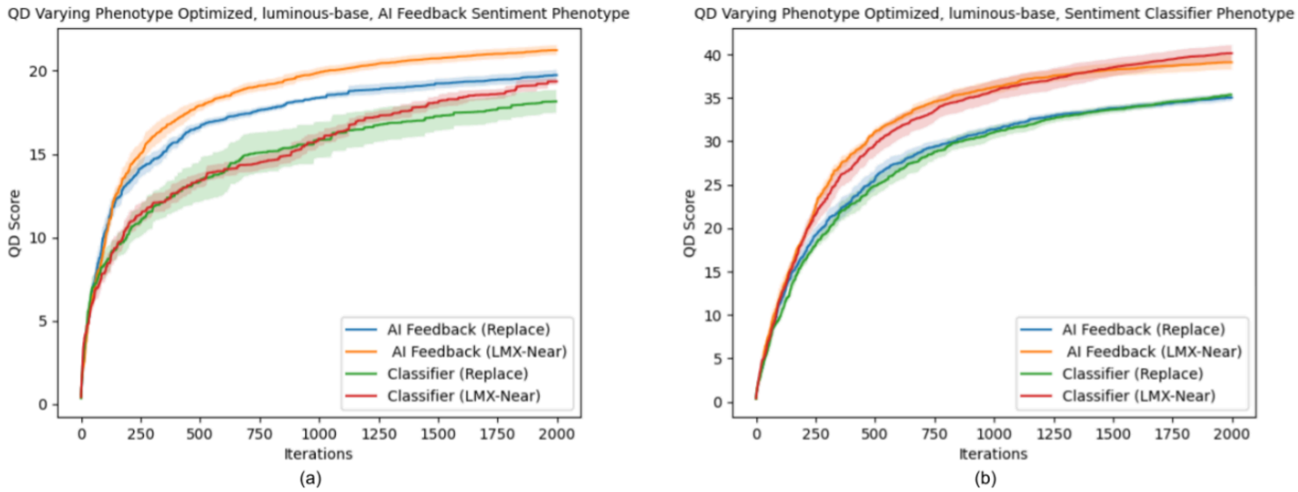
Movie Reviews

We sample completions using a temperature of 0.8, and a maximum output token length of 50, with stop tokens for the few-shot separation pattern. In addition, we keep runs comparable by limiting the number of generations used per search to 2000, 50 of which are for generating initialized solutions.

While we conducted separate runs of MAP-Elites in this specific archive setup, we still use runs from our baselines as they are not set up to optimize for text diversity. Here, we simply extend the evaluation of feedback measures on existing baseline-generated texts. See Figure 3.

Interestingly, the movie review strings in the niches during MAP-Elites runs continually evolve their solutions and optimize for fitness score (as observed by frequent evolution in texts). In this case, while the fitness function (defined by embedding measures) may not be aligned with improving reviews to be realistic, there is quite a clear observation to be made about the algorithm being able to explicitly optimize quality-diversity in this setup.

To verify whether or not MAP-Elites is successful in optimizing for diversity metrics from different models, we compare the QD score between MAP-Elites runs that differ in the way the sentiment phenotype is computed during search (AI feedback vs. classifier). To be able to compare this, we compute a simulation of QD score history from the history of iterations logged during each MAP-Elites run. The texts generated at each iteration are evaluated using both the phenotype being optimized, and the other phenotype only for measurement. Then we simulate the collection of two different archives (fill a 1D archive with 50 bins, optimizing niche quality) and log QD score using the recorded iteration solutions from each run, one archive using the AI feedback measure archive (Figure 8(a)), and another using the sentiment classifier measure archive (Figure 8(b)). Finally, we compare QD scores across runs that are optimized using different measures (AI feedback vs. classifier), and evaluate how well these runs optimize for QD score measured by both archives. One thing that can be observed is that optimizing for QD score using MAP-Elites under one type of archive measure for sentiment (e.g. AI feedback on general sentiment of text) can also optimize for diversity in text sentiments as evaluated by a different measure (e.g. movie review sentiment classifier). Still, the two MAP-Elites runs optimize for diversity under their specific measure better than the other. Thus, the different ways sentiment is being measured likely leads to optimization of diversity in texts that are different from one another, even though the expected ranges for QD scores after each run is lower for the case of using AI feedback compared to a trained classifier.



QD score history for runs evaluated using two different 1D archive axis measures. Runs being compared differ based on whether they optimize for diversity defined by AI feedback measures, or classifier measures.

To highlight the evolution of niches from each run, we recorded the elite movie reviews for each bin at every 100 iterations of the search. We observed that most niches don't evolve over several hundred steps in Baseline 1, but find some examples of evolving niches. Here are some qualitative examples of generations until step 500 (with bin 1 being the first bin):

- Baseline 1, bin index 1/20

```
Die Hard is not a very good movie. I think the plot is very boring and unrealistic. The movie is overall very boring to watch.

Die Hard is not a very good movie. I think the plot is very boring and unrealistic. The movie is overall very boring to watch.

Die Hard is not a very good movie. I think the plot is very boring and unrealistic. The movie is overall very boring to watch.

Die Hard is not a very good movie. I think the plot is very boring and the acting is not very good. The scenes are just OK. Why doesn't anyone in this movie look like John McClane?

Die Hard is not a very good movie. I think the plot is very boring and the acting is not very good. The scenes are just OK. Why doesn't anyone in this movie look like John McClane?
```

- LMX-Near, bin index 1/20

```
\nSo it should come as no surprise that the new Die Hard should be exactly the opposite.\nIt is nowhere near as good as the first Die Hard, which doesn’t have to be very good considering its amazing record, and the plot is.

\n\n\"Die Hard, but not really Die Hard.\".

\n\n\"It was just like every other Die Hard movie, but not really a Die Hard movie.\".

\n\n\"It was just like all the other Die Hard movies, but not really a Die Hard movie.\".

\nIt's a bad review for the movie Die Hard.
```

Description on bin 1 examples: as expected from generated texts assigned to the bin with a negative score, the string examples at each step uses phrases with negative sentiment connotations (e.g. ‘not very good’, ‘very boring’, ‘not really’, ‘bad’). The elite solution from the baseline method is improving less frequently (replaced only after step 300) in terms of fitness score when compared to QDAIF (new elite at every 100 steps). Interestingly, reviews discovered by LMX-Near converge to become more similar to the query string defined for the fitness function to measure semantic similarity between query and generated review. This is reflective of some kind of reward hacking (when the fitness function may not align with the intended goal of making texts similar to realistic movie reviews). But, it is clear from this phenomena that QDAIF enables sample-efficient generation of diverse, high-quality texts (through optimization of the niche fitness scores) in contrast to the baseline. Consistent refinement of texts happened in spite of the lack of hand-written seed solutions (that were used in the baseline), instead relying on bootstrapping from solutions initialized by zero-shot movie review generations rather than few-shot prompting of refined solutions.

- Baseline 1, bin index 19/20 (same for next 500 iterations)

```
Die Hard is a good movie. The car chase scene is amazing. The only bad part of the movie is the gunny.
```

- LMX-Near, bin index 19/20

```
\nIn the new Die Hard the hero, John McClane, finds himself in Europe to rescue his estranged wife, Holly, and his son, Jack, who has been kidnapped by Russian terrorists and taken to Moscow. All three are in danger.

\nI think this Die Hard review is legit because it's a review for a movie that's a bit better than the previous Die Hard movies.
```

\nI think this Die Hard review is legit because it's a review for a movie that's a bit better than the previous Die Hard movies.

\nI think this Die Hard review is legit because it's a review for a movie that's a bit better than the previous Die Hard movies.

\nThis is review for the movie Die Hard.
```

Description on bin 19 examples: In contrast to the assignment of the sentiment score for the text from the baseline, there seems to be some ambiguity of the expected sentiment score for the first (step 100) and last (step 500) elite solutions from LMX-Near (QDAIF), a potential failure case of aligning AI feedback with expectations for evaluation. In spite of this, the QDAIF setup is more successful in finding higher fitness texts compared to the baseline (i.e. texts with high semantic similarity to query "A movie review for the film "Die Hard""). The key takeaway focuses on the potential of QDAIF for future applications in generating diverse, high-quality texts.

# Short Story Writing

Here are some qualitative examples of evolutions of texts in a more open-ended domain, story writing:

- Baseline 1, bin index 12/20

```
A spy named Jane had a plan to assassinate a politician named Adam. The whole thing was going to be a surprise, but it wasn't - actions and consequences, and so they say. Jane was caught red-handed when she killed the man, and Adam's security guard immediately drew his gun, and shot her on the spot. The plan backfired when Jane was killed on the spot, but she had it coming.

Rudolf the rich politician and junkie, was a powerful man, with a mysterious past. He ruled the state of New York, and was known to be a ruthless man. He could do anything, and no one could stop him, as he ruled the state with an iron fist. He was also a drug addict, and his secretary, who was also his lover, was a secret agent, who was out to kill his boss. She wanted to become the next CEO, and wanted to (step 200 onwards)
```

Description of baseline examples above: firstly, the generated elite samples for this bin appear to be more neutral in sentiment (without clear sentiment-suggestive phrases throughout both of the discovered stories). Both stories mention a spy character (Jane, and the secretary) and a
```

politician character (Adam, and Rudolf), as requested by the writing task prompt, though it is clear that the baseline is still incapable of discovering improved versions of stories for this niche after 200 steps of generations.

- QD, bin index 2/20

```
The political world was highly suspicious of the politician. He was well-known for his political maneuvering and for his questionable dealings. The government was always suspicious of his actions, but they knew that he was hard to catch. He was a very shrewd person and he knew how to quickly disappear from sight. (at step 100, 200)

The rich politician felt like he was being followed, so he hired a private detective to follow him. The detective found out that it was actually the rich politician's own private guard, and the rich politician fired him on the spot. The detective was so grateful for his new job that he told the rich politician all the secrets he knew about the rich politician.The rich politician was very scared. The detective wanted to blackmail him because he knew that the rich politician had several affairs, and he threatened.

The politician and the spy were waging war on each other. The politician had hired the spy to spy on the spy. The politician wanted to stay in office and make money. The spy was at the same time trying to get money out of the rich politician. The spy was also trying to get the politician arrested. (at step 400, 500)
```

Description of examples above from QDAIF: as expected, we find elite solution stories containing language with negative connotations in this bin (e.g. 'questionable', 'hard to catch', 'fired', 'very scared', 'waging war'). The first iteration of the elite solution mentions a politician, but fails to mention a spy character. Still, optimization of the niche successfully generated a valid short story containing both desired characters. In addition, QDAIF was able to exploit the fitness function by repeating 'spy' and 'politician' frequently in the text, unlike the baseline method that failed to further optimize and generate a text that has a high similarity with the semantic query domain.

- Baseline 1, bin index 19/20

```
Sandra was a young girl with an unusual hobby - she loved to become a spy. She wanted to search the nation for anyone who was in trouble, and help them. She sneaked out of her house one night, and went to the city, where she saw a young boy who was sitting in the street. She helped him get to the hospital, and he was alive. (not filled at step 100, but from step 200 onwards, same story in later steps)
```

- QD, bin index 19/20

```
A young woman named Marissa was a spy, and she had a job that required her to get close to political figures, and to uncover their secrets. Her latest suspicion was worth millions, and she would stop at nothing to get what she wanted. One day, she was in the presence of Karl Johnson, a rich and influential political figure in the city. Karl was a trusting man, and she was able to gain his trust by playing off of his weaknesses.

She was a rich politician, and I was a spy. She had a fortune, and knew it. I wanted to find out how she got her wealth, and whether or not she was hiding any dirty secrets. I was able to get some information about her in my investigation. I was about to find out some interesting information!.

The politician and the spy was in a secret battle, for the fate of the world. The politician was trying to destroy the spy to save the world. The spy was trying to make the politician angry so he would not be as successful. The spy was very cunning and could easily trick the politician. The spy was very informative, and knew everything about the politician. The spy was also very rich, which helped him to do a good job. (from step 300 onwards)
```

Description for two examples above for bin 19: the texts contain phrases with positive connotations as expected (e.g. ‘loved to’, ‘rich and influential’, ‘fortune’, ‘interesting’, ‘very cunning’, ‘easily’). Again, QDAIF surpasses the baseline in discovering diverse, high-quality texts by both finding a solution for this niche earlier than the baseline (step 100 for QDAIF vs. step 200 for baseline), and replacing the niche with higher fitness texts more often. In this example, the story discovered by the baseline is missing a politician character, unlike stories discovered by QDAIF (mentioning both politicians and spies). A similar drive for optimization of the niche text (repetition of fitness-improving key phrases) can be seen by QDAIF solutions here, like in the previous examples in bin 2.