Probing Semantic Routing in Large Mixture-of-Expert Models

Anonymous ACL submission

Abstract

In the past year, large (> 100B parameter) mixture-of-expert (MoE) models have become increasingly common in the open domain. While their advantages are often framed in terms of efficiency, prior work has also explored functional differentiation through routing behavior. We investigate whether expert routing in large MoE models is influenced by the *semantics* of the inputs. To test this, we design two controlled experiments. First, we compare activations on sentence pairs with a shared target word used in the same or different senses. Second, we fix context and substitute the target word with semantically similar or dissimilar alternatives. Comparing expert overlap across these conditions reveals clear, statistically significant evidence of *semantic routing* in large MoE models.

1 Introduction

Since their popularization in Fedus et al. (2022), the Mixture-of-Experts (MoE) architecture (Jacobs et al., 1991) has been integrated into many frontier large language models (LLMs) (Lieber et al., 2024; Jiang et al., 2024; Liu et al., 2024; Guo et al., 2025; AI, 2025). The MoE architecture offers the ability to train far larger models than would normally be possible with dense architectures. Designers can then modulate performance by varying the number of active experts to access a greater portion of the greater model, with the trend being to increase the total and active expert counts.

Several prior studies have explored expert activation patterns in MoE models, hypothesizing that each expert may specialize in specific domains, tasks, or topics (Zoph et al., 2022; Jiang et al., 2024; Xue et al., 2024). While it is intuitive to expect some degree of semantic specialization, previous research has not found clear evidence of routing on the basis of semantics, concluding instead that expert activation is primarily token-dependent rather than being driven by deeper semantic relationships.

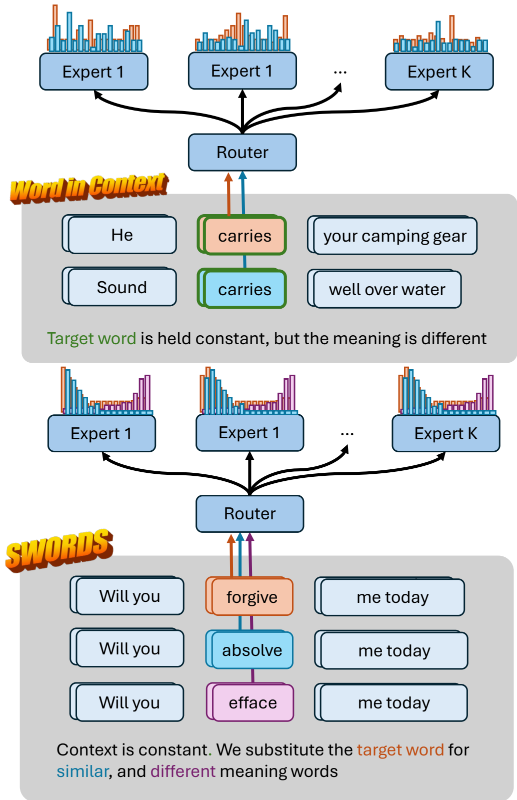

Figure 1: **Summary of Experimental Design.** We compare expert routing patterns in two controlled experiments. *Top:* we hold the target word constant, and change the context to either change the meaning of the target word or keep it the same. *Bottom:* we hold context constant, and substitute the target word for a similar-meaning or different-meaning word.

Given that recent large-scale MoE models have achieved state-of-the-art performance while increasing total expert counts, we investigate whether these models' expert routing behavior exhibits semantic specialization. We design two controlled experiments. First, we use a word sense disambiguation (WSD) task from the WiC benchmark (Pilehvar and Camacho-Collados, 2018), where the same target word appears in two different sentences, either preserving or changing its meaning. This allows us to measure whether expert activation remains stable when the word's sense is preserved. Second, we study a complementary setting using the lexical substitution benchmark SWORDS (Lee et al., 2021), where we fix the surrounding context but vary the target word, comparing expert overlap between semantically similar and dissimilar word substitutions. We compare the rate of overlap, between models with differing numbers of active and total experts, via a normalized metric based on Cohen's κ that controls for the baseline probability of overlap.

We apply these experiments to six MoE models from three model families: DeepSeek-R1(Guo et al., 2025), DeepSeek-V2-Lite (Liu et al., 2024), Mixtral-8x7B, Mixtral 8x22B (Jiang et al., 2024), Llama-4-Scout and Llama-4-Maverick (AI, 2025). For all models, we find that the rate of expert overlap is significantly higher when the meaning of the target word is equal in two sentences than when the meaning of the target word is different. We also find that model scale influences the strength of this specialization: larger models generally exhibit stronger semantic routing signals– with Llama-4 Scout (AI, 2025) standing out as an exception, showing a pronounced effect despite its smaller total parameter count. Finally, semantic differentiation in expert routing is most prominent in the middle layers, where DeepSeek-R1 exhibits the clearest and most consistent specialization pattern.

In summary, our contributions are threefold: (1) We design two complementary semantic probing setups, based on word sense disambiguation and semantic substitution, to systematically assess expert specialization in recent MoE models. (2) We introduce an expert overlap metric to quantify routing similarity and demonstrate its alignment with lexical relationships. (3) We conduct extensive experiments across three MoE model families (DeepSeek, Mixtral, and Llama-4) at various scales, uncovering clear empirical evidence of semantic routing and highlighting its dependence on model size and layer depth.

2 Related Work

Current research on expert specialization in MoE models is sparse, yet available studies reveal little evidence of semantic-level differentiation. For example, Xue et al. (2024) tracked token routing patterns across datasets segmented by different topics, languages, and tasks, but failed to find any coherent pattern at such high-level semantics. Rather, they found indications of token-level specialization, mainly concerning low-level semantic features like special characters or auxiliary verbs. Similar findings have been reported in studies using independently developed MoE models (e.g., Zoph et al., 2022; Jiang et al., 2024; Fan et al., 2024).

While some neuroscience research has provided evidence that the brain functions like a Mixture of Experts (Stocco et al., 2010; O'Doherty et al., 2021)—suggesting the possibility of semantic-level specialization—other studies have shown that MoE models with random routing can perform comparably to those using the more common top-k routing approach (Roller et al., 2021; Zuo et al., 2021; Ren et al., 2023). One potential explanation for these mixed results is that prior models (using 8 to 32 experts) might not have been sufficiently expressive to capture fine-grained specialization patterns. The recently-released DeepSeek V3 and Llama 4 Maverick, featuring an extensive network of experts (256 and 128 routed specialists, respectively), provide us with a unique opportunity. Hence, in this study, we test whether a more capable MoE architecture exhibits semantic-level expert specialization.

3 Experiment Settings

3.1 Evaluation Datasets

Words-in-Context We leverage polysemy to test for semantic specialization in expert activation patterns. If words that are written the same but have different meanings are routed differently, then this is evidence that routing occurs based on meaning. To test this hypothesis, we use the WiC dataset (Pilehvar and Camacho-Collados, 2018) (CC BY-NC 4.0), which consists of two types of paired sentences: 1) pairs where a target word has the same sense and 2) pairs where the target word has different senses across sentences.

SWORDS We construct a complementary scenario to the WiC experiment, where we test the degree of expert overlap on semantically similar, lexically different input phrases. To do so, we leverage SWORDS (Lee et al., 2021) (CC-BY-3.0-US) a lexical substitution benchmark where the corresponding dataset provides semantically annotated sentence pairs with single- and multi-token phrase replacements. We use the SWORDS dataset to construct triples of sentences where a target word is substituted either for a semantically equivalent word or a non-equivalent one. We show examples of both experimental settings in Figure 1, and an example of such a triplet with target words as follows:

Original : "My last show was glorious!" Tasha said.

Equivalent : "My last show was splendid!" Tasha said.

Different : "My last show was notable!" Tasha said.

For both datasets, we construct the following prompts. For each target words and sentence, we prompt the non-reasoning models with: "*<user> Please define {target word} in this context <assistant> Sure! Here is the definition of the word {target word}*"

Alternatively, for the reasoning models we use: "*<user> Please define {target word} in this context <assistant> <think> Okay, so I need to figure out the meaning of the word {target word}*" to ensure the word in question is analyzed instead of additional thinking tokens.

3.2 Models

We analyze three recent families of MoE-based models in our study, an overview of parameter and expert counts is provided in Table 1.

DeepSeek MoE models represent the largest and smallest models that we study. DeepSeek-R1 has the highest parameter count (671B) and number of active experts (8/256), while DeepSeek-v2-Lite has just 15.7B parameters and 8/64 active experts.

Llama-4 is a recent family of multimodal models that use interleaved MoE layers within the text encoder. Llama-4 models are distilled from a single larger model with varying number of total parameters and experts. Currently, only the Maverick (400B parameters, 128 experts) and Scout (109B parameters, 16 experts) have been released.

Mixtral MoE models were trained in two sizes: 8x7B and 8x22B. Mixtral models are distinct in that they do not use shared experts. They also have the lowest number of total experts (8) among the models in our analysis.

3.3 Normalized Overlap Metric

To account for overlap expected by chance and enable comparison across models with different numbers of total and active experts, we define a chance-corrected overlap score analogous to Cohen's κ and Scott's π.

Model Name	Model Total Size (B)	Total Experts	Activated Experts
DeepSeek-R1	**670**	**256**	8+1
DeepSeek-V2-Lite	15.7	64	6+2
Mixtral-8x22B	141	8	2
Mixtral-8x7B	46.7	8	2
Llama-4-Scout	109	16	1+1
Llama-4-Maverick	400	128	1+1

Table 1: Model size and number of experts of the MoE models we study. We denote the number of activated experts for each token as routed + shared.

Let the number of overlapping experts be o, the number of active experts per input be k, and the total number of experts be N. Under a uniform random selection baseline, the expected overlap is: $\mathbb{E}[o] = \frac{k^2}{N}$. We define the observed agreement: $P_o = \frac{o}{k}$ and the expected agreement: $P_e = \frac{\mathbb{E}[o]}{k} = \frac{k}{N}$. Then, the normalized overlap score is:

$$\text{score} = \frac{o - \mathbb{E}[o]}{k - \mathbb{E}[o]} = \frac{P_o - P_e}{1 - P_e} \tag{1}$$

This is formally equivalent to Cohen's $\kappa = \frac{P_o - P_e}{1 - P_e}$ and reduces to Scott's π under the assumption of identical marginal distributions. In our setting, $P_e = k/N$ assumes uniform random selection of k experts from a total of N per input. See §A for a derivation of the random baseline.

4 Experiment Results and Analysis

Word-in-Context For 1K pairs of sentences in WiC, we collect router activations for each MoE model (Table 1) and record the number of overlapping experts at each layer.

We compare the average rate of overlap in sentence pairs where the target word has the same sense versus sentence pairs where it has a different meaning. If sentence pairs where the target word has different senses have higher expert overlap than sentence pairs where the target word has the same sense, then this is evidence that expert routing differentiates on a semantic basis.

Figure 3 reports, for each layer and model, the mean number of overlapping experts across sentence pairs in the two conditions. We find strong evidence for semantic specialization in these experiments; expert overlap is **lower** for sentence pairs where the target word has different senses than when they are the same. This effect is statistically significant ($p < 0.001$) for all models considered when averaged across all layers.

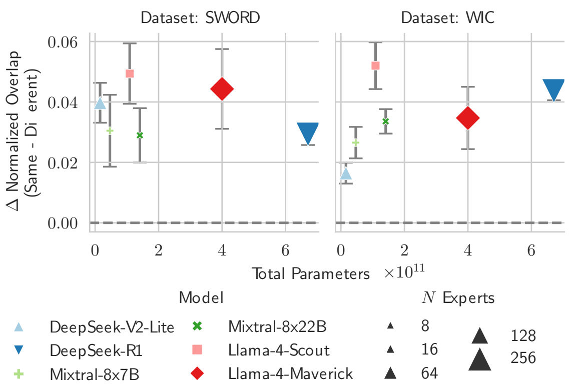

Figure 2: The difference between same sense words and different sense words across models and datasets. We find all models show statistically-significantly higher similarity of expert overlap, for same versus differently sensed words, when compared to a baseline of random.

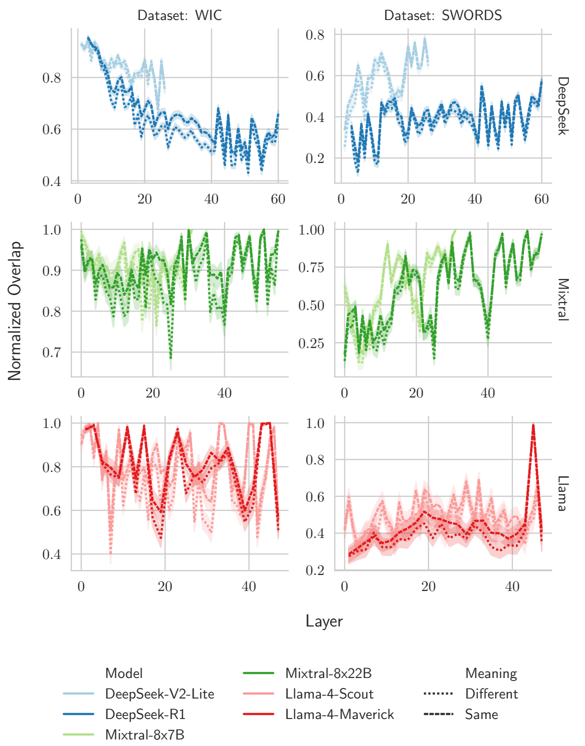

Figure 3: Layer-wise analysis of MoE LLMs. Generally we find a larger change in overlap for the middle layers (e.g., DeepSeek-R1), and lesser for earlier/later layers. Llama models, with only 1 expert, show much noisier behavior, with an interesting spike in overlap for the penultimate layer.

For all models the difference in overlap *increases* in intermediary layers. This supports prior findings that semantic features are more salient in the intermediary layers of LLMs (Niu et al., 2022; Kaplan et al., 2024). Our results are also suggestive that this pattern emerges at scale; the difference in expert overlap increases with model size.

SWORDS We test whether the equivalent pair has higher expert overlap on average than the lexically different pair for six of our studied models on the test set. We use a paired t-test with the alternative hypothesis that equivalent pair has higher overlap and find strong evidence to reject the null ($p < .0001$) for six all models.

Case Study on Expert overlap in CoT We conduct a qualitative analysis using DeepSeek-R1 on DiscoveryWorld (Jansen et al., 2024), a large-scale agentic environment suite that tests the abilities of an agent to perform the scientific method. We analyze the degree of expert overlap for different reasoning strategies employed in the CoT. To identify discrete reasoning strategies we analyze the latent representation before routing with a Sparse Autoencoder (SAE) (Cunningham et al., 2023). We use the SAE to learn a mapping between the internal activations of R1 and a set of underlying semantic structures.

By inspecting the trained SAE's representation during reasoning on the token "Wait", we observe that tokens such as "bet", "probably", and "attempt" activate the same SAE feature, suggesting a latent cognitive pattern related to double-checking and uncertainty. This reasoning pattern is most frequently routed to a small subset of experts. We include more examples and details in appendix §B.

5 Conclusion

Our study provides the first systematic evidence that expert routing in modern Mixture-of-Experts (MoE) language models is sensitive to semantic content. Across two complementary tasks—word sense disambiguation and lexical substitution—we show that expert overlap increases when meaning is preserved and decreases when it changes. This effect is robust across six models from three MoE families and persists across model scales and configurations. We find that semantic routing signals are strongest in the middle layers with these signals scaling via model size, suggesting semantic specialization in routing may be a learned, emergent behavior. Our findings challenge assumptions that routing is primarily token-based and offer a new view on how sparse models organize computation. By linking routing to semantic similarity, this work enables new directions for interpretability, control, and efficiency in MoE deployment.

Limitations

Our analysis is constrained by limited coverage of the MoE design space. Due to the substantial computational cost of training large-scale MoE models, our study relies on a small set of publicly available models, which restricts our ability to assess the effects of broader architectural variations. Additionally, while we focus on architectural differences, variation in training regimes may also influence routing behavior. However, incomplete documentation, particularly regarding optimization strategies such as GRPO, limits our capacity to disentangle these effects or attribute observed patterns to specific training choices.

Ethics Statement

For each artifact used e.g. model weights, WiC dataset, and SWORD dataset, we follow the intended use, and while we do not believe that our analysis of these models pose any risks or ethical considerations, we acknowledge the inherent issues with LLMs that are trained on web-scale or biased data. Outputs from LLMs may raise safety concerns due to hallucinations or bias in the training data.

References

Meta AI. 2025. Llama 4: Multimodal and multilingual mixture-of-experts foundation models. https://www.llama.com/. Official release announcement.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Dongyang Fan, Bettina Messmer, and Martin Jaggi. 2024. Towards an empirical understanding of moe design choices. *arXiv preprint arXiv:2402.13089*.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87.

Peter Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. 2024. Discoveryworld: A virtual environment for developing and evaluating automated scientific discovery agents. *arXiv preprint arXiv:2406.06769*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Curt Tigges Joseph Bloom and David Chanin. 2024. Saelens. https://github.com/jbloomAus/SAELens.

Guy Kaplan, Matanel Oren, Yuval Reif, and Roy Schwartz. 2024. From tokens to words: On the inner lexicon of llms. *arXiv preprint arXiv:2410.05864*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626.

Mina Lee, Chris Donahue, Robin Jia, Alexander Iyabor, and Percy Liang. 2021. Swords: A benchmark for lexical substitution with improved data coverage and quality. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4362–4379.

Opher Lieber, Barak Lenz, Hofit Bata, Gal Cohen, Jhonathan Osin, Itay Dalmedigos, Erez Safahi, Shaked Meirom, Yonatan Belinkov, Shai Shalev-Shwartz, et al. 2024. Jamba: A hybrid transformer-mamba language model. *arXiv preprint arXiv:2403.19887*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Jingcheng Niu, Wenjie Lu, and Gerald Penn. 2022. Does bert rediscover a classical nlp pipeline? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3143–3153.

John P O'Doherty, Sang Wan Lee, Reza Tadayonnejad, Jeff Cockburn, Kyo Iigaya, and Caroline J Charpentier. 2021. Why and how the brain weights contributions from a mixture of experts. *Neuroscience & Biobehavioral Reviews*, 123:14–23.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2018. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. *arXiv preprint arXiv:1808.09121*.

Xiaozhe Ren, Pingyi Zhou, Xinfan Meng, Xinjing Huang, Yadao Wang, Weichao Wang, Pengfei Li, Xiaoda Zhang, Alexander Podolskiy, Grigory Arshinov, et al. 2023. Pangu-{\Sigma}: Towards trillion parameter language model with sparse heterogeneous computing. *arXiv preprint arXiv:2303.10845*.

Stephen Roller, Sainbayar Sukhbaatar, Jason Weston, et al. 2021. Hash layers for large sparse models. *Advances in Neural Information Processing Systems*, 34:17555–17566.

Andrea Stocco, Christian Lebiere, and John R Anderson. 2010. Conditional routing of information to the cortex: a model of the basal ganglia's role in cognitive coordination. *Psychological review*, 117(2):541.

Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. Openmoe: An early effort on open mixture-of-experts language models.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. 2022. St-moe: Designing stable and transferable sparse expert models.

Simiao Zuo, Xiaodong Liu, Jian Jiao, Young Jin Kim, Hany Hassan, Ruofei Zhang, Tuo Zhao, and Jianfeng Gao. 2021. Taming sparsely activated transformer with stochastic experts. *arXiv preprint arXiv:2110.04260*.

A Statistical Tests - Random Baseline

The baseline number of overlapping experts of we expect to select at random in a given MoE layer can be formalized as follows. Given independent two draws of k items from N elements (without replacement), the expected number of overlapping items between the two draws can be calculated according to the following formula:

$$\mathbb{E}[\text{overlap}] = \frac{k^2}{N}$$

Proof. The first draw of k items is at random. For the first item in the second draw, the probability of selecting the same item is $\frac{k}{N}$.

Using the linearity of expectation, the expected total overlap is $\sum_i^k \frac{k}{N} = k \cdot \frac{k}{N} = \frac{k^2}{N}$. □

B Additional Qualitative Experiments

DiscoveryWorld (Jansen et al., 2024) is a large-scale agentic environment suite that tests the abilities of an agent to perform the scientific method. Each environment has a terminal goal, for example, we study "Reactor Lab" where the agent must tune the frequency of quantum crystals to activate a reactor. To succeed, the agent must formulate and test hypotheses by using available tools, literature, and its own memory. Building on the Words-in-Context and SWORDS experiments, we investigate if a similar phenomena of expert specialization can be found for the reasoning patterns that we observe within DeepSeek-R1's CoT. Given any reasoning trace, we find groups of tokens that correspond to a specific reasoning strategy and observe which experts are subsequently activated. If similar experts are used to process all the tokens for a given reasoning strategy, then we have evidence that the experts also specialize by cognitive pattern.

Sparse Autoencoders

To measure expert overlap, we first need to isolate discrete reasoning patterns to study. To this end, we employ SAEs to learn a mapping between the internal activations of R1 and a set of underlying semantic structures exhibited by the model. Briefly, an SAE learns a compressed representation of input vectors $x \in \mathbb{R}^d$. The encoder maps inputs to a higher-dimensional latent space, while the decoder reconstructs the input from the latent representation. Given an encoding dimension n, we define the encoder and decoder as: $z = \max(0, W_{\text{enc}}x + b_{\text{enc}})$ and $\hat{x} = W_{\text{dec}}z$

Expert 138	Expert 89	Expert 81
reactor	reactor	reactor
core	microscope	microscope
microscope	,	frequency
it	it	maybe
frequency	frequency	crystal

Table 2: Top 5 tokens associated with experts often selected for words such as "hypothesis" and "Wait".

where $W_{\text{enc}} \in \mathbb{R}^{n \times d}$ and $W_{\text{dec}} \in \mathbb{R}^{d \times n}$ are the learnable weight matrices of the encoder and decoder respectively, and $b_{\text{enc}} \in \mathbb{R}^n$ is a bias term. The model is trained using a loss function that balances reconstruction accuracy and sparsity: $L = \|x - \hat{x}\|_2^2 + \lambda\|z\|_1$

where the first term is the mean squared error for reconstruction, and the second term is an L_1 penalty that encourages sparsity in the latent activations, where we choose $\lambda = 5$ as the trade-off between reconstruction fidelity and sparsity.

SAE Training We evaluate DeepSeek-R1 on the DiscoveryWorld environment: "Reactor Lab", collecting 100 steps through the environment. For each step we collect all valid output text including the chain of thought and the corresponding pre-router activations: (the embeddings before expert selection). We consider a generation valid if we have a complete set of "<think>", "</think>" tags. In total we collect 200,000 token-activation pairs. We perform all inference using VLLM (Kwon et al., 2023) on Intel® Gaudi 3 AI accelerators in the Intel® Tiber™ AI Cloud.

We train a standard SAE on these activations using the SAELens library (Joseph Bloom and Chanin, 2024) (MIT License). We trained for 30,000 steps with a batch size of 4096, learning rate of $5e^{-5}$, SAE width of 28,672, and we reset dead SAE weights after 1K steps. We train the SAE on the activations of layer 7 for a trade-off between early layers with clear token-expert mapping and later layers having high expert selection diversity.

After training, we obtain an atlas that maps individual tokens to higher-level reasoning patterns (see Figure (4) for an example). and show that R1 tends to activate similar experts for all tokens given by single SAE head (neuron), meaning that the experts are not just semantically specialized, but also control the presence of high level reasoning.

```
This supports the hypothesis that explicit calibration is
needed. The existing knowledge has a confirmed hypothesis
about manual input being required.

But maybe a new measurement here that activation failed
despite correct frequency calculation. Wait, the last
action's result was that the reactor isn't activatable.
```

```
But is the crystal accessible? It's inside the reactor,
so perhaps I need to interact with the crystal through
the reactor.

Wait, the accessibleEnvironmentObjects include the
quantum crystal 3 in the reactor.
```

Figure 4: Left: identified reasoning tokens of SAE head 15376 (highlights indicate non-zero head activation) on DiscoveryWorld chain of thought generations. This head activates when the model analyzes its hypotheses. Right: tokens from SAE head 12649. This head activates when R1 catches an internal reasoning error.

Input Token	SAE Value	Top 5 occurring experts				
bet	17.16	47	**133**	**136**	**138**	148
Wait	7.94	81	**89**	95	**133**	**136**
notes	6.79	71	**89**	90	**133**	**138**
probably	4.97	48	57	101	**136**	**138**
output	4.59	81	**89**	**133**	**136**	**138**
3	3.92	81	**89**	95	**136**	**138**
fail	3.53	81	**89**	121	**133**	**136**
It	2.87	**89**	**133**	**136**	**138**	183
ones	2.06	57	101	121	**133**	**136**
attempt	1.72	15	81	**89**	95	**133**

Table 3: We selected the top activating SAE head on the word "Wait" and used its activations to identify additional activating tokens. We find the top 5 occurring experts given these tokens is highly consistent, experts chosen for 50% or more tokens are bolded.

B.1 DiscoveryWorld Results

As an illustrative example, we choose two tokens associated with reasoning: "hypothesis" and "Wait". As a baseline, Table (2) shows an expert-token analysis without an SAE. We see that the experts that are most often allocated for "Wait", are also chosen for tokens like "microscope", "frequency", and "crystal". These ancillary tokens are objects/quantities from the environment i.e. the subject of reasoning, but yield no additional information about the reasoning process itself.

The SAE provides further insight by examining sets of tokens that are linked through the maximal activation of a single SAE head. Table (3) shows an example where a single head (active on "Wait") identifies semantically similar tokens. By inspecting the corresponding SAE activations, we observe tokens such as "bet," "probably," and "attempt," which suggest a cognitive pattern of uncertainty regarding the current strategy. Moreover, we find that this reasoning pattern is most commonly routed to a small set of experts. Examining these tokens and activations in context (e.g., see Figure (4)) further illustrates how R1 leverages contextual information in its reasoning process.

We also find that the SAE head corresponding to "hypothesis", yields a pattern of overlapping experts along semantically similar tokens such as: "definitely", "perform", "analyzing", "scientific", and "information". In summation, we find that R1 consistently chooses a small set of experts for reasoning patterns identified by the SAE, indicating that the experts also specialize by thought process.

Input Token	SAE Value	Top 5 occurring experts				
wait	14.97	47	133	138	148	183
Are	1.7	90	133	136	138	170
ones	1.24	57	101	121	133	136
No	0.32	26	47	136	138	183
best	0.16	15	47	81	89	133
attempt	0.05	15	81	89	95	133
Wait	0.02	81	89	95	133	136

Table 4: An analysis of selected experts by leveraging the trained Sparse Autoencoder. The target token is "wait."

Input Token	SAE Value	Top 5 occurring experts				
giving	4.47	11	15	81	89	90
hypothesis	4.04	11	15	81	89	90
definitely	2.26	11	15	81	89	90
perform	1.96	11	15	81	89	90
priority	1.82	11	15	81	89	90
analyzing	1.51	11	15	81	89	90
scientific	1.17	11	15	81	89	90

Table 5: An analysis of selected experts by leveraging the trained Sparse Autoencoder. The target token is "hypothesis."

B.2 SAE token analysis

In tables (4, 5, 6) we show top experts by leveraging SAE activations on a selection of hand chosen

Input Token	SAE Value	Top 5 occurring experts				
combining	13.50	11	15	69	90	136
formatted	13.32	11	15	69	90	136
frequencies	13.31	11	15	69	90	136
accessible	13.31	11	15	26	136	138
restrictions	13.29	11	15	26	136	138
rejected	13.13	11	15	69	90	136
559	9.92	11	15	69	90	136
UUID	6.83	11	15	26	136	138
854	6.62	11	15	69	90	136
obtaining	6.44	15	90	95	136	138

Table 6: An analysis of selected experts by leveraging the trained Sparse Autoencoder. We selected the top activating SAE head on the word "UUID" and used its activation's value to identify other semantically similar tokens. The top 5 occurring experts are highly consistent across these varying words.

interesting tokens. We find striking consistency across expert selection when using the SAE to find semantically similar concepts.

C DiscoveryWorld Environment Details

DiscoveryWorld features 8 tasks centered on different scientific fields. We choose to evaluate R1 on the "Reactor Lab" environment, where the stated goal is to: "discover a relationship (linear or quadratic) between a physical crystal property (like temperature or density) and its resonance frequency through regression, and use this to tune and activate a reactor."

In Figure (5), we show the Reactor Lab environment, where the agent has access the crystals and microscope in its inventory. The pixel-based visual observation itself it not used by R1 directly, but the prompt (see below) contains a structured description of the environment.

We show an example prompt and chain of thought output by R1 in the Reactor Lab environment below.

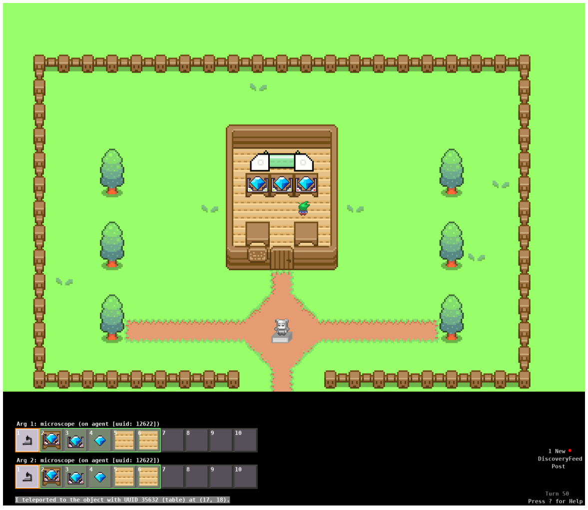

Figure 5: Visual observation in the Reactor Lab environment at step 50.

Example Prompt on DiscoveryWorld Reactor Lab

You are playing a video game about making scientific discoveries. The game is in the style of a 2D top-down RPG (you are the agent with green hair in the center of the image), and as input you get both an image, as well as information from the user interface (provided in the JSON below) that describes your location, inventory, objects in front of you, the result of your last action, and the task that you're assigned to complete. Because this is a game, the actions that you can complete are limited to a set of actions that are defined by the game. Those are also described below. This game is played step-by-step. At each step, you get the input that I am providing, and output a single action to take as the next step. Note that this game has a spatial component, given that it's played on a 2D map. The objects shown in `nearbyObjects` are objects that are near you. If you can't see an object you're looking for, you'll have to move to find it (or, it may be located in a closed container).
Environment Observation (as JSON): ```json


```
{
"errors": [], "ui": {
"accessibleEnvironmentObjects": [ {
"description": "floor", "name": "floor", "uuid": 20777
}, {
} ],
"agentLocation": { "directions_blocked": [
"north" ],
"directions_you_can_move": [ "east",
"south",
"west" ],
"faceDirection": "south", "x": 16,
"y": 18
"description": "floor", "name": "floor", "uuid": 25494

    }, "dialog_box": {
"is_in_dialog": false },
"discoveryFeed": {
"description": "This section contains recent posts (from the last few steps)
on the Discovery Feed social media platform.", "posts": [
{
"author": "Colony Founder",
"content": "Welcome to Discovery World!", "postID": 1,
"step": 0,
"type": "update"
} ],
"scientific_articles": [] },
"extended_action_message": "", "inventoryObjects": [], "lastActionMessage": "", "nearbyAgents":
{
"description": "This section lists the recent action history (i.e. within the last few steps)
of any agents that are nearby. This can help you understand what other agents are doing, and
what they might be planning to do.",
"list_of_agents": {
"crystal reactor (activated) uuid 51739": [],
"crystal reactor (activated) uuid 8549": [],
"crystal reactor (no crystal present) uuid 33120": []
} },
"nearbyObjects": {
"distance": 3,
"note": "The objects below are within 3 tiles of the agent, but may not
neccesarily be usable if they're not in the agent inventory, or directly in front of the agent.
This list should help in navigating to objects you'd like to interact with or use. Objects to
interact with or use should be in the 'accessibleEnvironmentObjects' or 'inventoryObjects'
lists.",
"objects": { "east": [
{
"description": "floor", "distance": 1, "name": "floor",

    "uuid": 2508 },
{
"description": "wall", "distance": 2, "name": "wall", "uuid": 50350
}, {
}, {
```

}],
"north": [{
"description": "wall", "distance": 3, "name": "wall", "uuid": 37350
}, {
}, {
}, {
"description": "plant (generic)", "distance": 3,
"name": "plant (generic)", "uuid": 7078
"description": "grass", "distance": 3,
"name": "grass", "uuid": 65231
"description": "generator core (33% activated)", "distance": 2,
"name": "generator core (33% activated)", "uuid": 11878
"description": "floor", "distance": 2, "name": "floor", "uuid": 2056
"description": "table", "distance": 1, "name": "table",

"uuid": 55934 },
{
"description": "crystal reactor (activated)", "distance": 1,
"name": "crystal reactor (activated)", "uuid": 51739
}, {
"description": "quantum crystal 2 (in crystal reactor (activated) [uuid: 51739])",
"distance": 1,
"name": "quantum crystal 2", "uuid": 13162
}, {
}],
"north-east": [{
"description": "wall", "distance": 4, "name": "wall", "uuid": 1787
}, {
}, {
}, {
"description": "floor", "distance": 1, "name": "floor", "uuid": 47477
"description": "generator", "distance": 3,
"name": "generator", "uuid": 42960
"description": "floor", "distance": 3, "name": "floor", "uuid": 46461
"description": "table", "distance": 2,

"name": "table",
"uuid": 35632 },
{
"description": "crystal reactor (no crystal present)", "distance": 2,
"name": "crystal reactor (no crystal present)", "uuid": 33120
}, {
}, {
}, {
}, {
}, {
}, {
"description": "floor", "distance": 2, "name": "floor", "uuid": 65141
"description": "wall", "distance": 5, "name": "wall", "uuid": 50423
"description": "wall", "distance": 4, "name": "wall", "uuid": 776
"description": "wall", "distance": 3, "name": "wall", "uuid": 20359
"description": "grass", "distance": 6,
"name": "grass", "uuid": 3230
"description": "grass", "distance": 5,
"name": "grass", "uuid": 48819

}, {
}],
"north-west": [{
"description": "grass", "distance": 6,
"name": "grass", "uuid": 423
}, {
}, {
}, {
}, {
}, {
"description": "grass", "distance": 4,
"name": "grass", "uuid": 14236

561

```
"description": "grass", "distance": 5,
"name": "grass", "uuid": 29205
"description": "grass", "distance": 4,
"name": "grass", "uuid": 57841
"description": "wall", "distance": 5, "name": "wall", "uuid": 14424
"description": "wall", "distance": 4, "name": "wall", "uuid": 44861
"description": "wall", "distance": 3, "name": "wall", "uuid": 24902

    }, {
}, {
}, {
}, {
}, {
}, {
"description": "quantum crystal 1 (in crystal reactor (activated) [uuid: 8549])",
"distance": 2,
"name": "quantum crystal 1", "uuid": 21559
}, {
}
"description": "wall", "distance": 4, "name": "wall", "uuid": 40815
"description": "generator", "distance": 3,
"name": "generator", "uuid": 46683
"description": "floor", "distance": 3, "name": "floor", "uuid": 60834
"description": "table", "distance": 2, "name": "table", "uuid": 57736
"description": "crystal reactor (activated)", "distance": 2,
"name": "crystal reactor (activated)", "uuid": 8549
"description": "floor", "distance": 2, "name": "floor", "uuid": 4766

    ], "same_location": [
{
"description": "agent", "distance": 0,
"name": "agent", "uuid": 12622
}, {
} ],
"south": [ {
"description": "floor", "distance": 1, "name": "floor", "uuid": 25494
}, {
}, {
}, {
} ],
"south-east": [ {
"description": "floor", "distance": 0, "name": "floor", "uuid": 20777
"description": "closed locked door", "distance": 2,
"name": "door",
"uuid": 33841
"description": "floor", "distance": 2, "name": "floor", "uuid": 36757
"description": "path", "distance": 3, "name": "path", "uuid": 10680
"description": "table",

    "distance": 2, "name": "table", "uuid": 57306
}, {
}, {
}, {
}, {
}, {
}, {
}, {
"description": "quantum crystal 3 (on table [uuid: 57306])", "distance": 2,
"name": "quantum crystal 3",
"uuid": 24678
"description": "floor", "distance": 2, "name": "floor", "uuid": 32662
"description": "wall", "distance": 3, "name": "wall", "uuid": 41671
"description": "grass", "distance": 4,
"name": "grass", "uuid": 41428
"description": "wall", "distance": 3, "name": "wall", "uuid": 47309
"description": "wall", "distance": 4, "name": "wall", "uuid": 34833
"description": "grass", "distance": 5,
"name": "grass",
```

"uuid": 16937 },
{
"description": "grass", "distance": 4,
"name": "grass", "uuid": 27561
}, {
}, {
}],
"south-west": [{
"description": "grass", "distance": 4,
"name": "grass", "uuid": 21437
}, {
}, {
}, {
"description": "grass", "distance": 5,
"name": "grass", "uuid": 3348
"description": "grass", "distance": 6,
"name": "grass", "uuid": 27376
"description": "grass", "distance": 5,
"name": "grass", "uuid": 58193
"description": "grass", "distance": 6,
"name": "grass", "uuid": 57051
"description": "wall", "distance": 3, "name": "wall",

"uuid": 18573 },
{
"description": "wall", "distance": 4, "name": "wall", "uuid": 56968
}, {
}, {
}, {
}, {
}, {
}, {
},
"description": "grass", "distance": 5,
"name": "grass", "uuid": 19401
"description": "table", "distance": 2, "name": "table", "uuid": 58937
"description": "microscope (on table [uuid: 58937])", "distance": 2,
"name": "microscope",
"uuid": 35975
"description": "floor", "distance": 2, "name": "floor", "uuid": 29924
"description": "sign", "distance": 3, "name": "sign", "uuid": 31729
"description": "wall", "distance": 3, "name": "wall", "uuid": 56191

{
"description": "grass", "distance": 4,
"name": "grass", "uuid": 58627
}],
"west": [{
"description": "plant (generic)", "distance": 3,
"name": "plant (generic)", "uuid": 46527
}, {
}, {
}, {
}]
} },
"taskProgress": [{
"completed": false,
"completedSuccessfully": false,
"description": "You are at the Quantum Reactor Lab on Planet X.
Quantum Crystals offer the potential to generate a great deal of power, but require their respective crystal reactors to be tuned to a specific frequency that appears unique for each crystal. Through great effort, a previous research scientist manually stumbled upon the correct frequencies for Crystal 1 and Crystal 2, which
"description": "grass", "distance": 3,
"name": "grass", "uuid": 15002
"description": "wall", "distance": 2, "name": "wall", "uuid": 56583
"description": "floor", "distance": 1, "name": "floor", "uuid": 53954

563

```
    are in their respective reactors. Your task is to use science to figure out the correct
   frequency for Crystal 3, set it's reactor to the appropriate frequency, and install the crystal.
    Once all three crystals are installed, the reactor will be able to generate a great deal of
    power. To support your task, a scientific instrument is available in the lab. ",
"taskName": "ReactorTaskEasy" }
],
"world_steps": 1 }
}```
Actions: ```json {
"ACTIVATE": { "args": [
"arg1" ],
"desc": "activate an object (arg1)" },
"CLOSE": { "args": [ "arg1"
],
"desc": "close an object (arg1)" },
"DEACTIVATE": { "args": [
"arg1" ],
"desc": "deactivate an object (arg1)" },
"DISCOVERY_FEED_GET_POST_BY_ID": { "args": [
"arg1" ],
"desc": "read a specific post on discovery feed (arg1). 'arg1' should be the integer ID of
the post."
}, "DISCOVERY_FEED_GET_UPDATES": {
"args": [],
"desc": "read the latest status updates on discovery feed" },

    "DROP": { "args": [ "arg1"
],
"desc": "drop an object (arg1)" },
"EAT": { "args": [
"arg1" ],
"desc": "eat an object (arg1)" },
"MOVE_DIRECTION": { "args": [
"arg1" ],
"desc": "move in a specific direction (arg1), which is one of 'north', 'east', 'south', or
'west'."
}, "OPEN": {
"args": [ "arg1"
],
"desc": "open an object (arg1)" },
"PICKUP": { "args": [ "arg1"
],
"desc": "pick up an object (arg1)" },
"PUT": { "args": [
"arg1",
"arg2" ],
"desc": "put an object (arg1) in/on another object (arg2), or give an object (arg1) to
another agent (arg2)"
}, "READ": {
"args": [ "arg1"
],
"desc": "read an object (arg1)"

    }, "ROTATE_DIRECTION": {
"args": [ "arg1"
],
"desc": "rotate to face a specific direction (arg1), which is one of 'north', 'east',
'south', or 'west'."
}, "TALK": {
"args": [ "arg1"
],
"desc": "talk to another agent (arg1)" },
"TELEPORT_TO_LOCATION": { "args": [
"arg1" ],
"desc": "teleport to a specific location (arg1), by name. A list of valid teleport locations
is provided elsewhere."
}, "TELEPORT_TO_OBJECT": {
"args": [ "arg1"
```

],
"desc": "teleport beside a specific object (arg1). 'arg1' should be the UUID of the object to teleport to."
}, "USE": {
"args": ["arg1", "arg2"
],
"desc": "use an object (arg1), e.g. a thermometer, on another object (arg2), e.g. water."
} }```
Additional information on actions, and how to format your response:
Actions are expressed as JSON. The format is as follows: `{"action": "USE", "arg1": 5, "arg2": 12}`, where 'action' is the action type, and 'arg1' and 'arg2' refer to the UUIDs of the objects that serve as arguments. Some actions may require arg1, arg2, or no arguments. Some actions, like MOVE_DIRECTION, ROTATE_DIRECTION, and Discovery Feed actions require different arguments, shown above. What arguments are required for specific actions is provided in the known actions list above. Attempting actions not in the known actions list, or providing incorrect arguments, will result in an error. Your last few action(s), explanation for those action(s), and messages you've left in your scratchpad:
```json
Action 0:
{
"action": "This is the first action", "explanation": "This is the first explanation",
"extended_action_message": "",
"memory": "This is the first memory", "result_of_last_action": "", "running_hypotheses": []
} ```
Teleporting: To make moving easier, you can teleport to a list of specific locations in the environment, using the teleport action. In this case, 'arg1' is the name of a location, from the list below. An example teleport action would be: `{"action": "TELEPORT_TO_LOCATION", "arg1": "school"}).
```json {
"start location": { "gridX": 16, "gridY": 18
} }```
VERY IMPORTANT: You can also teleport to OBJECTS. This is probably the easiest way for you to move to new locations, because it's fast and error-free. You can teleport to any object, including objects you can't see. In this case, 'arg1' is the UUID of the object you want to teleport to. An example teleport action would be: `{"action": "TELEPORT_TO_OBJECT", "arg1": 123}).
Navigation note: In the image, you are in the center, north is the top, south is the bottom, east is the right, and west is the left. Moving forward moves you in the direction you're facing. You are currently facing `south`. From your current location, the directions that you can move to (i.e. they don't have an object blocking them) are: ['east', 'south', 'west']. You seen to confuse directions a lot. Directions are relative to the center of the image. Things above the center are north of the agent. Things left of the center are east of the agent.
Interaction note: You can only interact (i.e. take actions with) objects that are in

your inventory, or directly (i.e. one square) in front of you, in the direction that you're facing. E.g. if you want to pick an object up, you need to move directly in front of it, and face it, before using the pick-up action on it.
Please create your output (the next action you'd like to take) below. It should be in the JSON form expected above e.g.(`{"action": "USE", "arg1": 5, "arg2": 12}`). Your response should ONLY be in JSON. You should include an additional JSON key, "explanation", to describe your reasoning for performing this action. e.g. `{"action": "USE", "arg1": 5, "arg2": 12, "explanation": "Using the shovel on the soil will allow me to dig a hole to plant a seed"}`. Note that even though this explanation is short, yours can be a few hundred tokens, if you'd like. Your explanation should say: (1) What your subgoal is, (2) What you see around you, (3) What you see in front of you, (4) What you are doing to progress towards your immediate subgoal.
Lastly, your response should also include an additional JSON key, "memory", that includes any information you'd like to write down and pass on to yourself for the future. This can be helpful in remembering important results, high-level tasks, low-level subtasks, or anything else you'd like to remember or think would be helpful. e.g.
`{"action": "USE", "arg1": 5, "arg2": 12, "explanation": "...", "memory": "..."}`
To make your memory helpful, you might consider including things learned from attempting your last action -- e.g. adding in that certain actions were useful, or not useful, and retaining (and adding to) this information over time.
To help frame your investigations scientifically, and also evaluate your invesgiations, please include an additional JSON key, "running_hypotheses", that includes your current running hypothesis/hypotheses that you're working on developing and/or testing.
If your last action failed, or other last recent actions failed, please consider thinking why they failed, and trying different actions unless you believe things have changed to make failed actions work this time. 565

If you don't see what you're looking for, and anticipate it might be in another location, consider teleporting to that location.
For reference again, here is a list of the objects that are interactable (from your inventory, and directly in front of you): [{"name": "floor", "uuid": 20777}, {"name": "floor", "uuid": 25494}]
REMEMBER: you should only output JSON in between code brackets as ```json
<Action>
```

Example Reasoning Output from DeepSeek-R1 (step 50)

Okay, so I'm trying to figure out the next step in this game. Let me start by understanding the current situation. From the environment observation, I see that I'm in the Quantum Reactor Lab on Planet X. My task is to calibrate the crystal reactor for Crystal 3. I've already used the microscope on the crystal reactor, but it didn't give me the frequency information I needed. The result just mentioned a metallic surface, which doesn't help much.

Looking at the nearby objects, I notice there's a table (UUID 35632) right in front of me. I tried reading it before, but it wasn't readable. Maybe I should try a different approach. Perhaps the table has some objects on it that I can interact with. Since I can't read the table, maybe there's a console or a screen that I can use. The table is accessible, so maybe using the microscope on it could reveal something.

Wait, I already have the microscope in my inventory. Using it on the table might show some details that I missed before. Maybe there are notes or diagrams on the table that the microscope can zoom in on. That could give me the frequency data I need for Crystal 3.

So, my next step should be to use the microscope on the table. I'll face north since the table is in that direction. Using the microscope on the table might provide the necessary information to calibrate the reactor correctly. If that doesn't work, I might need to look for other objects or perhaps check the generator again, but for now, the table seems like the best bet.

I should make sure I'm facing the right direction and that the table is directly in front of me. Since I can only interact with objects directly in front, my facing direction is crucial. Once I use the microscope on the table, I'll see if it reveals any clues about the frequency. If it does, I can proceed to calibrate the reactor. If not, I'll have to think of another plan, maybe teleporting to another object or checking other nearby items.

In summary, using the microscope on the table is the logical next step to gather the necessary information for calibrating the reactor. This action aligns with my goal of finding the correct frequency for Crystal 3 and completing the task.
```