SPD Autointerp Pipeline Report

Run: `wandb:goodfire/spd/runs/s-892f140b` **Model:** LlamaSimpleMLP (2 layers), trained on SimpleStories **Date:** 2026-02-06

Pipeline Overview

The SPD autointerp pipeline automatically generates and evaluates natural language descriptions of SPD parameter components. It consists of three stages:

1. **Harvest** -- Collect activation examples, token statistics, and correlations per component (GPU, SLURM)
2. **Interpret** -- LLM generates labels from harvested evidence (Gemini 3 Flash via OpenRouter)
3. **Eval** -- Intruder detection tests whether components are coherent without relying on labels

---

Interpretation Results

7,063 components interpreted across 12 module types and 2 layers.

Confidence	Count	Percentage
High	3,791	54%
Medium	2,031	29%
Low	1,241	18%

High-Confidence Examples

`h.1.mlp.c_fc:191` -- Verbs of physical or conceptual holding and preservation

> The component consistently triggers on verbs like 'hold', 'keep', 'stay', and 'hug' in both literal physical contexts (holding a hand) and metaphorical ones (holding secrets, keeping a memory alive). It demonstrates exceptionally high precision for predicting descriptors like 'tightly' and 'tight', indicating a specific role in processing the manner of retention or preservation in narrative arcs.

`h.0.attn.o_proj:280` -- Conjunctive transitions at sentence start within dialogue

> The component consistently fires on tokens like 'but', 'and', and 'at' immediately following an opening quotation mark and a sentence-ending punctuation mark. This suggests it tracks structural transitions inside character speech. The high PMI for predicted tokens like 'sure', 'remember', and 'maybe' further reinforces its role in facilitating conversational flow and character responses in stories.

`h.0.mlp.c_fc:87` -- Virtuous actions and pro-social character traits

> The component consistently activates on tokens describing communal virtues and selfless actions like 'help', 'kindness', 'generosity', 'trust', and 'honesty'. It also triggers on verbs describing favorable narrative resolutions such as 'saved', 'fixed', and 'thanked'. The predicted tokens suggest it helps the model anticipate descriptive suffixes and narrative continuation following these positive moral interactions.

`h.0.mlp.c_fc:747` -- The demonstrative pronoun 'this' preceding singular nouns and descriptors

> The component shows an extremely high recall (95%) and precision (75%) for the token 'this' as the current input. The activation examples consistently show it firing on 'this' when it introduces a noun phrase (e.g., 'this dream', 'this garden', 'this year'), which aligns with the predicted tokens like 'year', 'place', and 'magical'.

`h.1.attn.v_proj:237` -- Emotional heart reactions and physiological responses to stress

> The component triggers almost exclusively on the word 'heart' in the context of intense narrative emotions like fear, worry, or disappointment. It predicts verbs that specifically describe heart movement or emotional pain, such as 'sank', 'pounded', 'raced', and 'ached'. The activation examples consistently show characters reacting to sudden setbacks or threats.

`h.0.mlp.c_fc:850` -- Epistemic state verbs and narrative transitions in simple stories

> The component shows a strong recall for tokens like 'knew', 'looked', 'saw', and 'felt', which are essential for describing character internal states and perceptions in SimpleStories. The activation examples consistently highlight narrative pivot points where a character gains knowledge or experiences a shift in emotion or scenery. While precision is numerically low due to the extremely rare firing rate, the PMI and qualitative examples link it to thematic shifts and character realizations within simple narrative arcs.

`h.0.mlp.c_fc:609` -- emotional states and descriptions following possessive pronouns or 'felt'

> The component triggers on contexts following 'felt', 'my', or 'her' to project emotive or atmospheric adjectives like 'alive', 'powerful', or 'nervous'. While the specific 'Active tokens' in the examples are missing, the Recall and PMI metrics strongly indicate it facilitates internal state attributions (e.g., 'forest felt alive', 'her heart raced'). The extremely low activation rate suggests it is a highly specialized circuit for specific narrative-emotional transitions.

`h.0.mlp.down_proj:774` -- descriptive adjectives and suffix-based abstract nouns

> The component consistently triggers on a wide variety of descriptive adjectives such as 'soft', 'safe', 'deep', 'bright', and 'different'. Furthermore, it strongly predicts morphological suffixes like '##ness' and '##ly' as well as other characteristic word endings, suggesting it identifies qualitative descriptors and prepares for their adverbial or nominal extensions.

Medium-Confidence Examples

`h.1.mlp.down_proj:143` -- Character interaction following physical movement or approach verbs

> The component shows strong PMI with pronouns and characters ('he', 'she', 'lena') as input, and high PMI with physical actions ('knelt', 'arrived', 'approached', 'reached') as predicted outputs. While the activation rate is extremely low, the example contexts consistently feature a protagonist approaching an object or entity just before a shift in the story's action. The lack of highlighted tokens in the examples suggests it may fire specifically on the transition between an approach and its immediate consequence.

`h.1.attn.q_proj:16` -- visual descriptions of wonder and small-scale settings

> The component consistently correlates with tokens describing wide-eyed curiosity ('eyes', 'wide') and the opening of containers or rooms in small settings ('small town', 'quiet room', 'opened the box'). The predicted tokens emphasize descriptive adjectives like 'wide', 'cozy', and 'sparkling', suggesting a role in processing narrative moments of discovery or atmosphere. While the activation rate is extremely low, the PMI and precision metrics point toward a focus on descriptive sensory details and character reactions in simple stories.

`h.0.attn.k_proj:157` -- Sensory and mechanical verbs like ticking, creaking, and flowing

> The component shows strong PMI correlations with physical sounds (clock ticking, floor creaking) and natural movements (rivers/words flowing). While the activation examples do not highlight specific tokens, the predictive metrics strongly suggest a role in identifying contexts that precede these sensory descriptions. The very low activation rate indicates it is a highly specialized detector for these specific narrative motifs within the simple story domain.

`h.0.mlp.c_fc:203` -- abstract nouns and light imagery in narrative resolutions

> The component consistently correlates with tokens describing narrative atmosphere and emotional pivot points, such as 'light', 'surprise', and 'laughter'. While the activation examples do not highlight specific tokens with CI > 0.3, the high recall for 'light' and precision for specific names like 'Tim' suggests it may trigger on characters experiencing sudden moments of realization or wonder. The predicted tokens are largely morphological suffixes, indicating it may help structure the transitions between narrative description and dialogue or final story beats.

Low-Confidence Examples

`h.1.attn.q_proj:216` -- sub-word tokens and narrative transitions in 'banana'/night-time contexts

> The component fires extremely rarely, and its activation tokens in the provided examples are empty, making direct observation impossible. However, the PMI data suggests a strong association with sub-word fragments ('##an', '##ill', '##as') often found in 'banana' themed stories or time-based transitions ('one night', 'one evening'), and it shows an affinity for predicting surprise-related suffixes or punctuation marks common in SimpleStories narrative shifts; the low activation rate suggests it may handle highly specific edge cases of these patterns or is potentially miscalibrated.

`h.0.mlp.c_fc:284` -- narrative transition points and discovery verbs

> The component fires extremely rarely, making the activation rate almost negligible and the specific activation tokens absent in the provided examples. However, the PMI metrics show a strong preference for discovery and inquiry verbs like 'found,' 'asked,' 'set,' and 'rec' (likely 'recreate' or 'recognize'), while predicted tokens like 'laimed' (exclaimed) and 'herself' point toward character-driven narrative shifts and dialogue beats typical of story resolutions or plot advancements. Due to the lack of clear, highlighted active tokens in context, the pattern remains speculative based on token correlations alone.

`h.1.attn.k_proj:214` -- Character decisions and natural scenery elements

> The component exhibits extremely low firing rates and lacks any active tokens in the provided story examples, suggesting it might be a 'dead' or highly specific feature. While the PMI data points toward verbs of decision-making (decided, agreed) and natural nouns (sky, tree, bush), the absence of direct activations makes a definitive interpretation impossible. It potentially represents an K-projection signal for specific nouns following character actions, but the evidence is too sparse for high confidence.

Example: Full Prompt and Response

Below is a complete example showing what the LLM sees and how it responds.

Component: `h.1.mlp.down_proj:849`

Prompt (truncated)

```
Label this neural network component from a Stochastic Parameter Decomposition.

## Background

SPD (Stochastic Parameter Decomposition) decomposes a neural network's weight matrices into
"subcomponents". Each subcomponent has a causal importance (CI) value predicted *per sequenc
by a small auxiliary neural network. CI indicates how necessary the component is for the mod
at that position: high CI (close to 1) means the component is essential and cannot be ablate
(close to 0) means it can be removed without affecting output. The training objective encour
sparsity: as few components as possible should have high CI for any given input.

## Model Context

**Model**: simple_stories_train.models.llama_simple_mlp.LlamaSimpleMLP (2 layers)
**Dataset**: SimpleStories is a dataset of 2M+ short stories (200-350 words each) at a grade
The stories cover diverse themes (friendship, courage, loss, discovery) and settings (magica
schools, forests, space). The vocabulary is simple, everyday English. Stories feature common
elements: characters with names, emotions, dialogue, and simple plot arcs with resolutions.

## Component Context

**Component location**: mlp.down_proj in layer 2 of 2
**Activation rate**: 1.11% (fires on ~1 in 89 tokens)

---

## Correlations with Input Tokens

The following metrics concern correlations between this component firing and the "current" t

**Recall:** _"What % of this component's firings occurred on token X?"_
  'was': 44%
  'is': 12%
```

```
  'be': 8%
  'were': 5%
  'not': 4%
  'are': 3%
  's': 2%
  'just': 2%
  'the': 2%
  'became': 2%

**Precision:** _"When token X appears, what % of the time does this component fire?"_
  'is': 86%
  'was': 70%
  'be': 59%
  'became': 55%
  'been': 47%
  'are': 47%
  'were': 38%
  'not': 26%
  '##eed': 25%
  'become': 21%

**PMI:** _Tokens with higher-than-expected co-occurrence_
  ' is': 4.00
  ' was': 3.79
  ' be': 3.63
  ' became': 3.55
  ' been': 3.39
  ' are': 3.38
  ' were': 3.19
  ' not': 2.80



---

## Correlations with Predicted Tokens

The following metrics concern correlations between this component firing and the token the m

**Precision:** _"When the model predicts token X, what % of the time is this component activ

**High (70-90%)**: 'happening', 'possible'

**Medium (50-70%)**: 'easy', 'wrong', 'yours', 'worth', 'mine', 'important'

**PMI:** _Tokens with higher-than-expected co-occurrence_
  ' happening': 4.03
  ' possible': 3.83
  ' easy': 3.78
  ' wrong': 3.72
  ' yours': 3.71
  ' worth': 3.67
  ' mine': 3.60
  ' important': 3.46
```

```
  ' hers': 3.44
  ' tricky': 3.44


---

## Activation Examples

_Showing tokens where CI > 0.3 (component is active)_

Ex 1: "and sadness. she decided that even though her friend was not there, she could still c
  Active tokens:

Ex 2: ". the love he had was lost, and mia was just a reflection of what he longed for. as"
  Active tokens: "was" (1.00), "was" (0.35)

Ex 3: "silent, and the laughter that once filled the air was now just a memory. he felt resp
  Active tokens: "was" (1.00), "now" (0.60)

Ex 4: "the captain held up the chicken. " but it is a very funny chicken! just look at it!"
  Active tokens: "is" (0.79), "very" (0.47)

Ex 5: "the dark hallway, mia found a door that was slightly open. she pushed it and saw"
  Active tokens:
...
```

Response

```
{
  "label": "State-of-being verbs and copular linking predicates",
  "confidence": "high",
  "reasoning": "The component consistently triggers on forms of the verb 'to be' (is, was, b
}
```

Intruder Detection Eval

Tests component coherence by showing an LLM 4 real activating examples + 1 intruder from a different component. The LLM must identify the odd one out. Random baseline = 20%.

100 components scored (10 trials each)

Metric	Value
Mean accuracy	57.2%
Median accuracy	60.0%

Metric	Value
Random baseline	20.0%

Score Distribution

Coherence Level	Count	Bar
Incoherent (<20%)	1	
Weak (20-40%)	10	▓▓▓▓▓
Moderate (40-60%)	33	▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓
Good (60-80%)	45	▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓
Excellent (80-100%)	11	▓▓▓▓▓

56% of components score 60%+ (Good or Excellent), indicating the majority of SPD components learn coherent, distinguishable activation patterns.

Pipeline Architecture

```
spd-harvest <wandb_path> --n_gpus 8 --n_batches 1000
    => harvest/{run_id}/activation_contexts/   (examples, token stats)
    => harvest/{run_id}/correlations/           (co-firing, PMI)

spd-autointerp <wandb_path> --cost_limit_usd 50
    => autointerp/{run_id}/results_*.jsonl      (labels per component)

python -m spd.autointerp.eval.scripts.run_intruder <wandb_path> --limit 100
    => autointerp/{run_id}/eval/intruder/       (coherence scores)

python -m spd.autointerp.scoring.scripts.run_label_scoring <wandb_path> --scorer detection
    => autointerp/{run_id}/scoring/detection/   (label accuracy scores)
```

Key Features Built Today

- **Cost tracking with budget limits** across all LLM-calling pipelines
- **Robust retry handling** for OpenRouter transient failures (8 retries, exponential backoff)
- **Resume support** everywhere (append-only JSONL, skip completed on restart)
- **Forced JSON schemas** via OpenRouter's `response_format` for structured outputs
- **Three evaluation methods:**
- Intruder detection (label-free component coherence)

- Detection scoring (label predictiveness)
- Fuzzing scoring (label specificity)

Generated by Claude Code on 2026-02-06. Total API cost for this run: ~$12 interpretation + ~$0.50 intruder eval.