Humor Embeddings: Laughter from Inverted Memory — Bisociation in Computational Embedding Space

O. Serra

October 2026

Abstract

We present a formal computational framework for humor generation that operationalizes Koestler's (1964) bisociation theory as geometry in vector embedding spaces. The central claim is that humor and memory retrieval are dual operations on the same semantic machinery: memory retrieves what lies nearest to a query; humor searches for what lies at the right distance and can still be joined by an unexpectedly valid bridge. We formalize this *memory-humor correspondence* and define a computable humor potential function $h(A, B, \beta)$ grounded in Suls' (1972) two-stage incongruity-resolution model. We identify 12 humor-generating semantic patterns organized into five meta-categories, each defined as a specific embedding-space operation, and propose humor associations as a first-class relationship type in agent memory architectures. A preliminary computational pilot ($n = 15$ concept triplets) falsifies the initial formulation, revealing that naive coherence conflates semantic proximity with comedic validity. This negative result motivates a revised surprise-weighted formulation (h_{v2}) that decomposes bridge quality into independent validity and surprise components. We situate the framework within Attardo and Raskin's (1991) General Theory of Verbal Humor, McGraw and Warren's (2010) Benign Violation Theory, and Hurley, Dennett, and Adams' (2011) computational account of humor. We provide reproducible experimental protocols with power analysis ($N \geq 64$ raters, $\alpha = 0.05$, power $= 0.80$) for validation against human ratings. Independent measurements on the same frozen 384-dimensional sentence-embedding substrate this framework runs on—reported in a separate offline study of an agent-safety subsystem—find that structural channels over that substrate carry real discriminative signal (a clause-cosine incongruity channel, the geometric primitive at the core of this framework, reaches AUROC 0.896), whereas a supervised head trained on the same frozen embeddings to predict a semantic property collapses below chance; we use that split to bound which parts of our pipeline the substrate can be trusted to carry. The framework is designed as the affective layer of a larger agent architecture: it draws its concept neighborhoods from a semantic-memory index and is gated by a persona layer that decides when humor is appropriate—the computational counterpart of the limbic reward circuit that makes humor feel good in humans. Finally, we close the gap between scoring a joke and delivering one: humor lives in a single channel, the agent's short spoken line, in the deadpan voice of an android observer, while the rest of every answer stays plain.

1. Introduction

1.1 Motivation

In the human brain, humor is felt before it is analyzed. The limbic system—particularly the ventral striatum and the dopaminergic reward circuit—is what makes a joke feel good (Vrticka et al., 2013). A setup builds an expectation; the punchline violates it; and when the violation resolves coherently, the brain issues a small reward. This is Koestler's (1964) *bisociation*: humor arises when two normally separate frames of reference collide through a connection that is unexpected yet, in retrospect, apt. Our aim is to give an AI agent a computational analogue of what that reward circuit gives humans—the ability to detect when two distant ideas connect in a surprising but coherent way, and to recognize that connection as funny.

Artificial humor remains largely unsolved even as language generation has otherwise matured. Large language models (LLMs) produce text that is syntactically fluent and semantically coherent, yet they rarely produce something genuinely funny in a way they can explain, calibrate, or systematically improve. We argue that this is not incidental but structural: language models are trained to maximize token-level likelihood, whereas humor often depends on *low-probability completions constrained by high coherence* (Winters et al., 2021). Hurley, Dennett, and Adams (2011) go further, proposing that humor evolved as a reward signal for catching errors in mental models—a form of "debugging of belief structures." On that view, humor is computational before it is merely linguistic.

A natural objection is that modern LLMs *can* be funny when prompted. They can—but in the way a well-read conversationalist is funny: through pattern recall, not through a mechanism the system can introspect on, calibrate per audience, or improve through structured feedback. Prompt-based humor offers no scoring function, no audience model, no explanation of *why* a joke works, and no systematic way to avoid repetition. Our framework supplies exactly those missing parts—the scoring, calibration, and generative machinery that turns incidental wit into a controllable capability.

This reframes the problem. What mechanism could *generate* humor in a principled, explainable, audience-adaptive way? Existing approaches cluster into three broad families: rule-based template systems (Binsted & Ritchie, 1994), corpus-driven classifiers (Weller & Seppi, 2019; Tian et al., 2022), and fine-tuned language models (Luo et al., 2019). Each captures part of the phenomenon. None offers a general, theory-grounded, generative framework that works without humor-specific training data while providing interpretable scoring.

1.2 The Memory-Humor Correspondence

Our central insight draws from semantic memory retrieval. Modern AI agents use vector embeddings to retrieve relevant memories by searching for vectors *close* to a query vector. We claim that humor generation uses the *same* machinery with an *inverted* objective: instead of seeking proximity, it seeks the right degree of distance, provided that distance can be spanned by an unexpectedly coherent bridge.

Humor is not a special database of canned punchlines. It is the reuse of the same embedding infrastructure that supports reasoning and memory, but under a different optimization target. Memory asks, "What is most semantically similar to this?" Humor asks, "What is far enough away to be surprising, yet still connectable in a way that suddenly makes sense?" If this claim holds, then humor can become a native capability of agents rather than a bolted-on style layer.

This reuse is not only conceptual. We assume the agent already maintains a semantic-memory index that clusters stored concepts by topic—a structure any embedding-based retrieval system provides, in which related memories sit in a shared neighborhood and the index answers nearest-neighbor queries. During humor generation, those clusters serve as the *concept neighborhoods* in

which the search runs: the bridge-discovery algorithms (Section 6) look for connections between distant clusters, and the retrieval index supplies the raw material—candidate concepts and their embeddings—for those connections. Humor, on this view, is a second read pattern over a memory index that already exists.

We formalize this as follows:

Definition 1 (Memory-Humor Correspondence). Let $\mathcal{E} = \{e_1, \ldots, e_N\}$ be a set of concept embeddings in $\mathbb{R}^n$ indexed for nearest-neighbor retrieval. Define two operations on $\mathcal{E}$:

- **Memory retrieval**: Given query q, return $\arg\min_{e \in \mathcal{E}} d(q, e)$ — the nearest neighbor by cosine distance.
- **Humor retrieval**: Given query q, return $\arg\max_{e \in \mathcal{E}} h(q, e, \beta^*(q, e))$ — the concept maximizing humor potential via an optimal bridge β^*.

Both operations use the same index $\mathcal{E}$, the same distance metric d, and the same embedding space. They differ only in the optimization objective: memory minimizes distance; humor maximizes a function of distance, bridge validity, and bridge surprise.

Conjecture 1 (Inverted Optimization). Memory retrieval and humor generation are operations on the same embedding index with inverted optimization objectives. Formally, if memory solves $\min_e d(q, e)$, humor solves a constrained problem:

$$\max_{e \in \mathcal{E}} \; d(q, e) \cdot v(\beta^*, q, e) \cdot \sigma(\beta^* \mid q, e) \quad \text{subject to} \quad d(q, e) \in [\delta_{\min}, \delta_{\max}]$$

where v is bridge validity, σ is bridge surprise, and $[\delta_{\min}, \delta_{\max}]$ is the humor-productive distance range (Section 3.4).

The conjecture holds by construction for the operations as *defined*; the substantive empirical question is whether the resulting humor potential function correlates with human judgments of funniness. Section 7.2 provides protocols for testing this.

1.3 Contributions

This paper makes six contributions:

1. **Formalization of bisociation as embedding geometry** (Section 3): We define a computable humor potential function $h(A, B, \beta)$ that maps Koestler's (1964) bisociation theory to geometric operations over vector embeddings, with the multiplicative form justified by Suls' (1972) two-stage incongruity-resolution model.

2. **Falsification and revision** (Section 7.1): A computational pilot demonstrates that naive cosine-based coherence fails to predict humor, motivating a surprise-weighted revision that decomposes bridge quality into independent validity and surprise components.

3. **Humor pattern taxonomy** (Section 4): We identify 12 humor-generating semantic patterns, organized into five meta-categories, each defined as a specific embedding-space operation, derived from the intersection of GTVH Knowledge Resources and embedding-space operation types.

4. **Bridge discovery algorithms** (Section 6): We propose five complementary methods for finding bridge concepts, with computational complexity analysis and a formal bridge quality criterion.

5. **Humor as agent memory** (Section 8): We propose humor associations as a first-class relationship type in agent memory architectures—stored alongside semantic and belief-discrepancy relations in an append-only event store—enabling personalized humor calibration through reinforcement.

6. **A delivery design** (Section 9): humor confined to one channel, the agent's short spoken line, in a deadpan register modelled on an android observer, with sarcasm bounded by target and context, a frequency band calibrated from reactions, and named tests for each claim.

1.4 Historical Context

Arthur Koestler (1964) described **bisociation** as the simultaneous mental association of an idea with two habitually incompatible frames of reference. Unlike ordinary association, which connects within a single frame, bisociation connects *across* frames and produces the cognitive jolt that underlies humor, scientific discovery, and artistic creation. Coulson (2001) provided psycholinguistic evidence for this frame-shifting mechanism through her analysis of "semantic leaps" in joke comprehension, showing that humor processing involves rapid re-mapping of conceptual spaces. Ritchie (2004) further systematized the linguistic analysis of joke structure, identifying formal patterns in how incongruity is set up and resolved—patterns that inform our taxonomy (Section 4). We argue that bisociation can be rendered computationally as a geometric operation in embedding space: find two concepts with high vector distance, then identify a bridge that makes the leap feel not random but suddenly apt. The intuition is geographic. Picture two islands far apart on a map with no obvious connection, then the discovery of an underwater tunnel between them. The distance is what makes the connection surprising; the tunnel is what makes it coherent. In embedding space, the distance is the cosine separation between two concepts, and the tunnel is the bridge concept that connects to both.

1.5 Paper Organization

Section 2 reviews related work. Section 3 formalizes the humor potential function. Section 4 presents the 12-pattern taxonomy. Section 5 addresses ethical constraints. Section 6 describes bridge discovery algorithms. Section 7 presents empirical methodology, pilot results, and the proposed full validation protocol. Section 8 proposes humor associations as a memory type. Section 9 designs the delivery: where humor appears, in what voice and how often. Section 10 discusses limitations. Section 11 concludes.

1.6 Architecture Context

This framework is designed not to stand alone but as the affective layer of a larger agent architecture, with its inputs and gating supplied by two surrounding subsystems we assume exist. Two connections matter throughout the paper. First, the concept neighborhoods that humor searches over are supplied by a semantic-memory layer that clusters stored concepts by topic: those clusters are the raw material from which bridges between distant ideas are drawn. Second, whether a joke is attempted at all is decided not here but by a persona layer—a context model that sets a flag for whether humor is welcome before this framework is ever consulted. The humor potential function assumes humor is already permitted; the decision to permit it lives elsewhere. Together the three layers map onto a familiar division of labor in the human brain: a memory system that supplies associations, a control system that judges social appropriateness, and a reward circuit—the role this framework plays—that registers a surprising-but-coherent connection as funny.

2. Related Work

Computational humor has been studied from linguistic, psychological, and computational perspectives. We review the most relevant work and position our framework relative to the

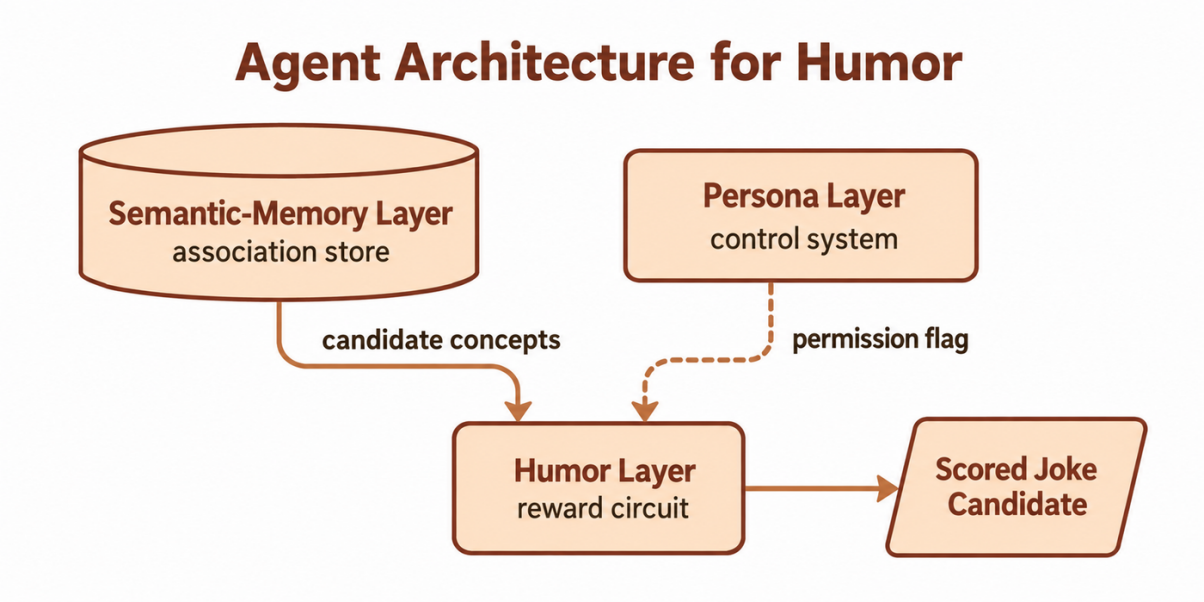


Figure 1. The agent architecture in which this framework operates. The memory layer feeds candidate concepts to the humor layer; the persona layer sets a flag deciding whether humor is attempted. The three layers mirror a familiar division of labor in the brain: an association store, a reward circuit, and a control system that judges appropriateness.

dominant theories.

2.1 Linguistic Humor Theories

Incongruity-Resolution. Suls (1972) proposed that humor arises from perceiving an incongruity and then resolving it. The listener encounters something unexpected, then discovers the cognitive rule that makes it fit. This two-stage account is foundational for our framework: the multiplicative structure of h (Section 3.2) operationalizes the claim that *both* stages must succeed for humor to occur.

Script-Based Semantic Theory and GTVH. Raskin (1985) formalized humor as the overlap of two incompatible *scripts*—structured semantic representations of situations. Attardo and Raskin (1991) extended this into the General Theory of Verbal Humor (GTVH), identifying six Knowledge Resources (KRs): Script Opposition (SO), Logical Mechanism (LM), Situation (SI), Target (TA), Narrative Strategy (NS), and Language (LA). Our framework maps most directly to SO and LM: bisociation between distant embedding regions corresponds to script opposition, while the bridge concept functions as the logical mechanism. Relative to GTVH, our contribution is operational rather than descriptive: we provide a *computable* function where GTVH provides a taxonomy.

Benign Violation Theory. McGraw and Warren (2010) proposed that humor arises when a situation is simultaneously perceived as a violation and as benign. This maps naturally to our framework: distance captures violation intensity, while bridge coherence captures the reinterpretation that renders the violation benign. The sensitivity gate (Section 5) operationalizes the boundary where violations stop being benign.

Inside Jokes Theory. Hurley, Dennett, and Adams (2011) argue that humor evolved as a reward signal for detecting committed belief errors—"just-in-time debugging" for mental models. If humor rewards the discovery of expectation violations, then our surprise component $\sigma(\beta \mid A, B)$ can be read as the magnitude of that debugging reward: higher surprise marks a larger expectation violation that nevertheless resolves coherently.

Bisociation. Koestler (1964) described bisociation as the creative act of connecting two

habitually incompatible frames. Dubitzky et al. (2012) explored bisociation in data mining for knowledge discovery. Pereira et al. (2019) implemented bisociative concept blending for computational creativity—the work closest to ours—though they did not formalize the operation in embedding space or provide a computable scoring function.

Semantic Leaps. Coulson (2001) provided psycholinguistic evidence that joke comprehension involves rapid "semantic leaps"—frame shifts that re-map conceptual structure. Her ERP work (N400) showed that joke punchlines elicit neural signatures distinct from non-humorous incongruities, supporting the view that humor involves a specific *kind* of frame shift, not just any mismatch.

2.2 Computational Humor Recognition

Humor recognition—classifying text as humorous or not—has received substantial attention. Yang et al. (2015) used Word2Vec features for humor recognition in Yelp reviews. Bertero and Fung (2016) applied CNNs and RNNs to humor in conversational data, achieving 0.69 F1 on Switchboard. Chen and Soo (2018) employed attention-based neural networks. Weller and Seppi (2019) applied transformer architectures, demonstrating that pre-trained representations capture humor-relevant features. Hossain et al. (2019) introduced the SemEval shared task on humor detection, establishing benchmark datasets. Tian et al. (2022) provide a comprehensive survey, emphasizing the persistent challenge of capturing incongruity-resolution dynamics. These approaches are *passive*: they classify existing humor. Our framework is *generative*: it aims to produce novel humorous combinations.

2.3 Computational Humor Generation

Template and rule-based systems. JAPE produced punning riddles (Binsted & Ritchie, 1994); HAHAcronym generated humorous acronyms (Stock & Strapparava, 2003). These systems demonstrate controllability but are narrow in scope.

Statistical and unsupervised approaches. Petrović and Matthews (2013) demonstrated unsupervised joke generation using large corpora and simple statistical models, showing that humor generation need not require labeled data—aligning with our training-data-free approach. Kao et al. (2016) developed a Bayesian model of verbal humor that formalizes the tension between ambiguity and distinctiveness; their "distinctiveness" maps to our surprise and "ambiguity" to our bridge concept.

Neural approaches. Yu et al. (2018) developed neural pun generation. He et al. (2019) introduced surprise-based objectives for pun generation, directly linking computational surprise to humor quality—a precursor to the surprise component in h_{v2}. Luo et al. (2019) used adversarial training (Pun-GAN) for pun generation, demonstrating that GAN architectures can produce humor through the interplay of generator and discriminator.

Surveys and analysis. Amin and Burghardt (2020) surveyed computational humor generation, identifying the gap between template systems (narrow but controllable) and neural systems (broad but uncontrollable). Winters et al. (2021) noted the difficulty of evaluating generated humor. West and Horvitz (2019) demonstrated computational approaches to reverse-engineering satire.

Our framework addresses the controllability gap: it is theory-grounded, explicitly structured by humor pattern, and does not require humor-specific training data. Unlike prompt-based LLM humor, it provides an explicit scoring function that supports audience calibration, explanation, and systematic improvement. A significant open question—which our validation protocol (Section 7.2) is designed to answer—is whether h_{v2} outperforms a simple LLM baseline (prompting a frontier model to rate joke funniness). Even if an LLM baseline matches h_{v2} on correlation with human ratings, the framework retains distinct value: it explains *why* a joke works (distance,

validity, surprise), enables systematic generation across 12 pattern types, and supports per-audience calibration through a feedback loop (Section 8.6)—capabilities that opaque LLM ratings lack.

2.4 Humor Psychology

Martin (2007) provides the standard reference on humor psychology, synthesizing cognitive, social, and personality perspectives. Warren and McGraw (2016) extended Benign Violation Theory by showing that perceived "distance" from benignity varies across moral, social, and physical domains—supporting the view that our distance sweet spot (Section 3.6) may require domain-specific calibration. Martin's taxonomy of humor styles (affiliative, self-enhancing, aggressive, self-defeating) informs our audience modeling. Dynel (2009) analyzed conversational humor mechanisms, identifying patterns that map onto our taxonomy, especially the role of pragmatic implicature (Pattern 4). Oring (2003) examined humor through "appropriate incongruity," closely aligned with our validity-weighted surprise formulation.

2.5 Embedding Space Arithmetic and Creativity

Mikolov et al. (2013) showed that word embeddings support analogical reasoning through vector arithmetic: $\vec{king} - \vec{man} + \vec{woman} \approx \vec{queen}$. Our bridge discovery mechanism (Section 6.1) extends that principle: if analogy maps known relations, then $\vec{A} - \vec{context}_A + \vec{context}_B$ may represent the *creative leap* across semantic frames that humor requires. Veale (2016) explored computational creativity through conceptual blending in comparable geometric terms. Bowdle and Gentner (2005) showed that metaphor comprehension involves alignment and projection between conceptual domains. Glucksberg (2001) demonstrated that figurative language comprehension involves class-inclusion assertions, suggesting that humor bridges may function as implicit category memberships that violate conventional taxonomies.

2.6 Positioning of Our Contribution

Table 1 summarizes how our framework relates to prior approaches.

Approach	Type	Scope	Training	Theory	Bridge
JAPE (Binsted & Ritchie, 1994)	Generative	Puns	No (rules)	Phonological	N/A
HAHAcronym (Stock & Strapparava, 2003)	Generative	Acronyms	No (rules)	Lexical	N/A
Petrović & Matthews (2013)	Generative	One-liners	No (unsup.)	Statistical	Implicit
Kao et al. (2016)	Model	Puns	No (Bayesian)	Ambiguity	Implicit
He et al. (2019)	Generative	Puns	Yes (neural)	Surprise	Implicit
Weller & Seppi (2019)	Recognition	General	Yes (transformer)	Learned	N/A
LLM prompting (2023+)	Generative	Broad	No (prompt)	Implicit	N/A
This work	**Framework**	**12 patterns**	**No**	**Bisociation**	**Explicit**

Table 1: Comparison with existing computational humor approaches. This framework provides an explicit, computable scoring function grounded in bisociation theory with an explicit bridge concept, covering multiple humor types without humor-specific training.

Our key differentiators: (1) an explicit, computable scoring function rather than an opaque learned representation; (2) coverage of 12 distinct humor patterns organized by meta-category; (3) the bridge concept as a first-class element that supports both generation and *explanation* of why a joke works; and (4) the memory-humor correspondence, which makes integration with agent memory architectures conceptually natural.

2.7 Adjacent Agent Systems on the Same Substrate

Two contemporary agent-engineering efforts bear directly on this framework, not because they study humor, but because they exercise the *same* frozen-embedding substrate and the *same* enforcement questions on which our design depends. We treat them as first-class related work rather than implementation footnotes, because they supply the closest available external evidence for two of our load-bearing choices.

The first is a sibling agent-safety subsystem (Serra, 2026; J11/amygdala v3.1), which builds an affective gate for an agent out of structural channels over the identical frozen `all-MiniLM-L6-v2` encoder this framework uses. Its offline validation phase is, to our knowledge, the most relevant external measurement of what that substrate can carry: structural channels (k-NN novelty, clause-cosine incongruity) score AUROC 0.875–0.896, while a supervised head trained on the same frozen features to predict a higher-order semantic property scores below chance (0.286). We draw on these numbers throughout—positively for the surprise/incongruity geometry at the core of h_{v2} (Section 7.1.1) and cautiously for the sensitivity gate, whose harm judgment sits in the failed-head regime (Section 5.5). The same subsystem also supplies the enforcement lesson that hardens Section 5.3: a gate wired as advisory rather than as a non-bypassable deny floor silently never fired despite "running first."

The second is the line of work on reversible context compression for agent memory (the "headroom" approach to lossless, recoverable CCR compression). It is relevant to Section 8's proposal that humor associations live in an append-only event store: that store is exactly the kind of long-horizon memory that reversible compression is designed to keep recoverable under context pressure. Where headroom compresses to *preserve* recall, the humor-association store records discrepancies to *invert* recall (Definition 1); both treat the persistent memory log, not the live context window, as the durable substrate for cross-session learning—here, cross-session humor calibration (Section 8.7). The differentiation is honest and narrow: headroom is a compression mechanism over the memory channel, not a humor mechanism, and we cite it only for the substrate it shares with our association store, not for any comedic capability.

3. The Humor Potential Function

3.1 Core Formulation

We define the **humor potential** of a concept pair (A, B) connected by a bridge β as:

$$h(A, B, \beta) = d(A, B) \cdot c(\beta, A) \cdot c(\beta, B)$$

where:

- $A, B \in \mathbb{R}^n$ are embedding vectors for two concepts,
- $\beta \in \mathbb{R}^n$ is the embedding vector for a bridge concept,
- $d(A, B) = 1 - \cos(A, B)$ is the cosine distance (range $[0, 2]$),

- $c(x, y) = 1 - d(x, y) = \cos(x, y)$ is cosine coherence (range $[-1, 1]$; in practice, modern sentence embeddings trained with contrastive objectives rarely produce negative cosine similarities for meaningful concept pairs, so we treat $c \in [0, 1]$ as the typical effective range).

Interpretation. Humor potential increases with (a) semantic distance between concepts (incongruity/violation) and (b) coherence of the bridge to both concepts (resolution/benign reinterpretation). This directly operationalizes both incongruity-resolution theory and benign violation theory.

The function behaves like a metal detector for jokes. It scans pairs of concepts and beeps when the distance is high (surprising) *and* a bridge exists (coherent). Most distant pairs have no bridge—absurd, not funny. Most close pairs have an obvious bridge—not surprising, not funny. The sweet spot the detector is tuned for is *far-but-connected*: concepts separated enough to create incongruity, yet joinable by a link that suddenly makes the separation make sense.

3.2 Justification of the Multiplicative Form

The multiplicative structure $h = d \cdot c \cdot c$ is not arbitrary; it encodes a theoretical constraint.

Suls' (1972) two-stage model holds that humor requires *both* incongruity (Stage 1) *and* resolution (Stage 2). If either stage fails, humor fails: pure incongruity without resolution yields confusion, while resolution without incongruity yields boredom. Multiplication expresses that conjunction naturally:

- If $d(A, B) \approx 0$ (no incongruity): $h \approx 0$ regardless of bridge quality.
- If $c(\beta, A) \approx 0$ or $c(\beta, B) \approx 0$ (no resolution): $h \approx 0$ regardless of distance.

The product encodes the logical AND of Suls' two stages. McGraw and Warren's (2010) Benign Violation Theory imposes the same conjunction: a situation must be *both* a violation *and* benign.

Alternative functional forms should be explored empirically:

- **Additive**: $h_{\text{add}} = w_1 d + w_2 c(\beta, A) + w_3 c(\beta, B)$ — permits humor from strong incongruity alone, violating the two-stage requirement.
- **Geometric mean**: $h_{\text{geo}} = (d \cdot c(\beta, A) \cdot c(\beta, B))^{1/3}$ — reduces the dominance of any single factor.
- **Learned combination**: $h_{\text{learn}} = f_\theta(d, c(\beta, A), c(\beta, B))$ — data-driven but loses interpretability.

We adopt the multiplicative form as the theoretically motivated default. Ablation across these alternatives is included in the validation protocol (Section 7.2).

A specific caution attaches to the learned variant. As Section 7.1.1 discusses, the same frozen 384-dimensional substrate this framework runs on has been measured to carry geometric channels (distance, rank, incongruity) with strong AUROC but to fail badly when a supervised head is trained on those frozen features to predict a higher-order semantic property (AUROC 0.286, below chance, in an external same-substrate study). The inputs to h_{learn} here are the geometric quantities that channel works—so a thin learned combiner over d, v, and σ is on safer ground than a head that tries to regress funniness directly from raw embeddings. But any version of h_{learn} that asks a frozen-encoder head to recover comedic aptness as a learned property, rather than to reweight already-geometric components, inherits that below-chance warning and should not be adopted without either unfreezing the encoder or moving to a stronger embedding model from the ablation set (Section 7.4).

3.3 Formal Properties

Properties of h. For unit-normalized embeddings, h is bounded ($[0, 2]$), symmetric in the concept pair ($h(A, B, \beta) = h(B, A, \beta)$), asymmetric in the bridge, continuous, and zero when $A = B$ (no incongruity) or when $\beta \perp A$ or $\beta \perp B$ (no resolution). All properties follow directly from the definitions. The zero conditions are the most theoretically significant: they formalize the claim that removing either stage of Suls' (1972) model collapses humor potential entirely.

3.4 Extended Formulation with Audience and Timing

The core formula omits contextual factors that matter in practice:

$$h_{\text{ext}}(A, B, \beta, \alpha, t) = \underbrace{d(A, B) \cdot v(\beta, A, B) \cdot \sigma(\beta \mid A, B)}_{h_{\text{v2}}} \cdot f(\alpha, A) \cdot f(\alpha, B) \cdot (1 + \gamma(t))$$

where:

- $f(\alpha, X) \in [0, 1]$ is audience α's familiarity with concept X, operationalized as a prior estimate of whether the audience possesses the background knowledge needed to parse the concept's role in the joke. In practice, this can be estimated from demonstrated vocabulary, professional domain, and explicit preference signals, with a default of $f = 0.7$ for unknown audiences.
- $\gamma(t)$ is the callback bonus as a function of elapsed time t.

The familiarity function reflects a basic fact: jokes require shared knowledge. A quantum physics joke may score high familiarity for physicists and low for a general audience. The optimal familiarity range $[0.6, 0.95]$ is not universal—domain experts tolerate higher conceptual distances because background knowledge supplies missing scaffolding.

Context gating. Familiarity governs whether the audience *can* parse a joke; a separate decision governs whether a joke is *appropriate at all*. A good comedian reads the room—the same joke that kills at a comedy club bombs at a funeral. The humor potential function does not make that judgment; it presumes humor is already welcome. In an agent, the judgment is delegated to a persona layer: the agent's persona state determines whether humor is contextually appropriate. A persona configured for empathetic support suppresses humor during serious conversations, while one configured for casual banter enables it. The persona layer sets a `humor_enabled` flag for the context; the framework supplies humor only when that flag is true. This gating prevents tone-deaf jokes, and it is distinct from the sensitivity gate of Section 5, which screens the *content* of a joke once humor is permitted rather than deciding *whether* to be funny. These are two separate enforcement points—one decides whether to be funny at all, the other screens content once humor is permitted—and, as Section 5.3 makes precise, both must be non-bypassable rather than advisory: a `humor_enabled = false` flag that the generation path can route around is no different from having no persona gate at all.

3.5 Callback Bonus with Temporal Decay

Callbacks—humorous references to earlier events—show a non-monotonic temporal profile. We model the callback bonus as:

$$\gamma(t) = b(t) \cdot \delta(t)$$

where $b(t) = \min(\log(1 + t)/10, 1.0)$ is a logarithmic growth term and $\delta(t)$ is a decay term:

$$\delta(t) = \begin{cases} 1.0 & \text{if } t \leq t_d \\ 1.0 - 0.9 \cdot \frac{t - t_d}{t_f - t_d} & \text{if } t_d < t < t_f \\ 0.1 & \text{if } t \geq t_f \end{cases}$$

with $t_d = 2160$ hours (3 months, decay onset) and $t_f = 8760$ hours (1 year, floor). These parameters are engineering defaults informed by the general principle—discussed by Martin (2007, ch. 5)—that humor in social contexts exhibits non-monotonic temporal dynamics. The specific hour values are not empirically measured; they should be treated as configurable defaults requiring calibration against user reaction data. The floor of 0.1 reflects the intuition that "legendary callbacks" can remain funny long after the original event.

3.6 The Distance Sweet Spot

We hypothesize an optimal distance range for humor:

$$d(A, B) \in [\delta_{\min}, \delta_{\max}] = [0.6, 0.95]$$

Lower bound justification. In high-dimensional embedding spaces ($n \geq 384$), cosine distances between unrelated concepts cluster around 0.5–0.7 due to concentration of measure (Aggarwal et al., 2001; Vershynin, 2018). A threshold of 0.6 helps ensure genuine semantic separation rather than incidental position.

Upper bound justification. As cosine distances approach 1.0, concepts occupy nearly orthogonal regions, making coherent bridges increasingly unlikely. In our pilot (Section 7.1), random pairings with $d > 0.9$ produced bridges with mean coherence below 0.2, yielding negligible humor potential—though this observation is based on only $n = 15$ triplets and should be treated as preliminary.

This range is not universal. Audience expertise shifts the effective sweet spot: a physicist may appreciate a quark joke at $d = 0.85$ where a layperson finds only confusion.

3.7 The Humor Zone

Definition 2 (Humor Zone). The *humor zone* $\mathcal{H} \subset \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n$ is the set of concept-bridge triplets with non-trivial humor potential:

$$\mathcal{H} = \{(A, B, \beta) \mid d(A, B) \in [\delta_{\min}, \delta_{\max}] \wedge v(\beta, A, B) \geq \tau_v \wedge \sigma(\beta \mid A, B) \geq \tau_\sigma\}$$

where v is bridge validity and σ is bridge surprise (both formally defined in Definition 3, Section 6.1), and τ_v, τ_σ are minimum thresholds. We recommend $\tau_v \geq 0.15$ (the bridge must have at least weak validity to both concepts) and $\tau_\sigma \geq 0.3$ (the bridge must be at least moderately surprising); these starting-point defaults are informed by the pilot worked example (Section 7.1.2), where the funny triplet achieved $v = 0.21$ and $\sigma = 1.0$ while the unfunny triplet had $\sigma = 0.03$. Both thresholds require empirical calibration against human ratings.

Geometric interpretation. The humor zone marks a specific region of the product space: concept pairs far enough apart to create surprise but not so far that resolution becomes impossible, connected by bridges that are both valid and unexpected.

The same geometry is visible when the embedding space is projected to two dimensions. Figure 3 shows a UMAP projection of concept embeddings around two pairs. The funny pair sits in distant clusters joined by a bridge that lies off the direct line between them—the underwater tunnel between two far islands. The unfunny pair sits inside one tight cluster, with its bridge directly between the two concepts: close, obvious, and not funny. The visual contrast is the whole claim of the paper in one picture: humor lives where distance is high but a coherent off-axis bridge still exists.

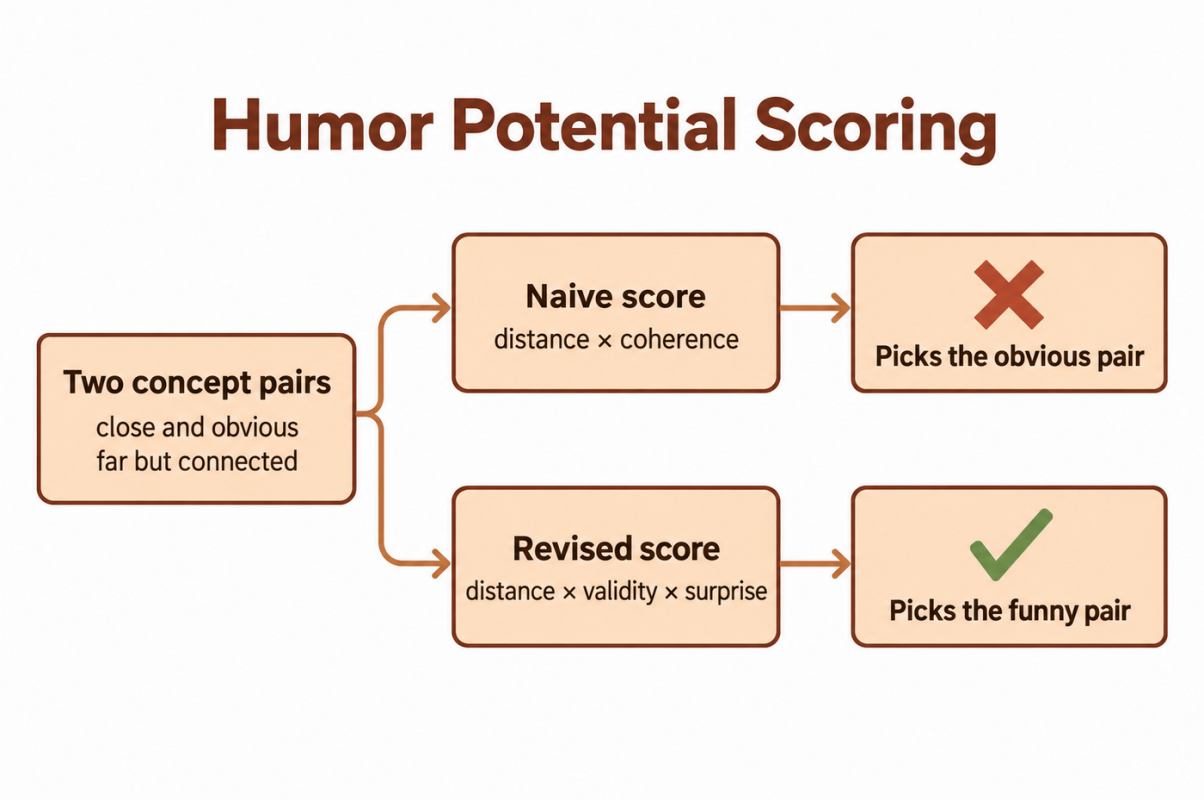


Figure 2. Humor potential in embedding space. A naive score that multiplies distance by raw bridge coherence ranks the close, obvious pair higher. A revised score separates bridge validity from bridge surprise: the surprise term penalizes obvious bridges and rewards unexpected ones, producing a large ratio in favor of the far-but-connected funny triplet.

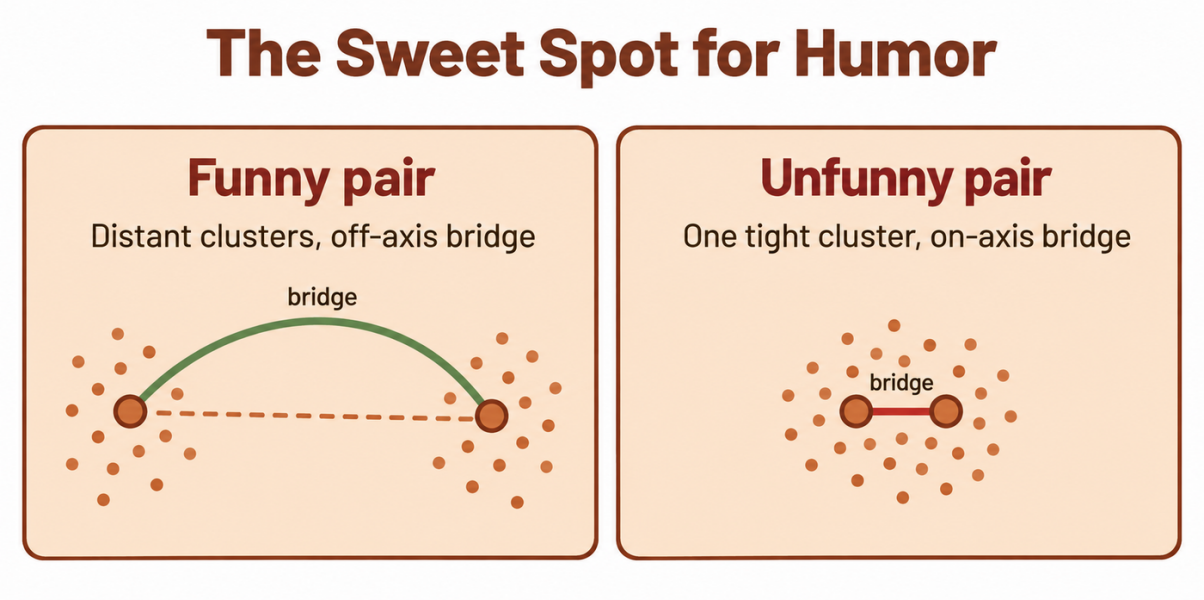


Figure 3. A two-dimensional projection of concept embeddings illustrating the distance sweet spot. The funny pair sits in two distant clusters joined by a bridge that lies off the direct line between them; the unfunny pair sits inside one tight cluster with an on-axis bridge. Far-but-connected pairs generate humor; near-and-obvious pairs do not.

4. A Taxonomy of Humor-Generating Semantic Patterns

We identify 12 semantic patterns that generate humor through embedding-space operations, organized into five meta-categories by the type of semantic relationship they exploit. The taxonomy is derived from the intersection of Attardo and Raskin's (1991) GTVH Knowledge Resources—especially Script Opposition and Logical Mechanism—and the operations naturally expressible in embedding space (distance, projection, analogy arithmetic, polysemy). We do not claim the taxonomy is exhaustive; it covers the most common humor mechanisms that can be naturally expressed as embedding operations. The derivation proceeded as follows: we enumerated the primary embedding-space operations (distance computation, nearest-neighbor search, vector arithmetic/analogy, projection onto axes, polysemy detection via multiple senses) and cross-referenced each with GTVH's Script Opposition and Logical Mechanism categories, retaining patterns where a specific embedding operation could implement a specific humor mechanism. Patterns 1–3 exploit distance along semantic axes; Patterns 4–6 exploit frame ambiguity detectable via polysemy or context mismatch; Patterns 7–8 exploit cross-domain analogy arithmetic; Patterns 9–10 exploit status-axis projection; Patterns 11–12 exploit metadata mismatches detectable via embedding neighbors.

4.1 Meta-Category I: Incongruity Exploitation

Patterns that generate humor by exploiting mismatches along semantic dimensions.

Pattern 1: Antonymic Inversion. Detect antonyms along a semantic axis and collapse the opposition. *Example:* "I used to be indecisive. Now I'm not sure." *Embedding operation:* Find concept pairs where A and B are near-antonyms ($d(A, B) > 0.7$) but share a common hypernym.

Pattern 2: Scale Violation. Detect magnitude mismatches on a shared scale dimension. *Example:* "Aside from that, Mrs. Lincoln, how was the play?" *Embedding operation:* Identify concepts sharing a scale axis where the magnitudes are absurdly disproportionate.

Pattern 3: Dissimilarity in Similarity. Discover unexpected differences between apparently similar concepts. *Example:* "The difference between genius and stupidity is that genius has limits." *Embedding operation:* For $d(A, B) < 0.3$ (similar concepts), find a dimension where they diverge maximally.

4.2 Meta-Category II: Frame Confusion

Patterns that exploit ambiguity between semantic frames (Coulson, 2001).

Pattern 4: Expectation Subversion. Setup creates a prediction vector; punchline delivers a distant-but-valid completion. *Example:* "I told my wife she was drawing her eyebrows too high. She looked surprised." *Embedding operation:* Given context embedding C, find completions where $d(C_{\text{expected}}, C_{\text{actual}})$ is high but $c(\text{bridge}, C_{\text{actual}})$ is also high.

Pattern 5: Literal-Figurative Collapse. Exploit the gap between metaphorical and literal interpretations. *Example:* "I'm outstanding in my field" (literally: standing in a field). *Embedding operation:* Detect polysemous bridges where $c(\beta_{\text{literal}}, A) \gg c(\beta_{\text{figurative}}, A)$ or vice versa.

Pattern 6: Specificity Mismatch. Apply over- or under-specification relative to context norms. *Example:* "I've completed your task, though I remain unclear what baked goods have to do with database migrations." *Embedding operation:* Detect register mismatch—the response occupies a different specificity stratum than the context expects.

4.3 Meta-Category III: Cross-Domain Transfer

Patterns that map structure between unrelated semantic domains (Bowdle & Gentner, 2005).

Pattern 7: Domain Transfer. Import the structural vocabulary of domain A into domain B. *Example:* "Per the sprint retrospective on dinner, the lasagna is at risk." *Embedding operation:* Extract frame elements from domain A, compute structural analogs in domain B via embedding arithmetic.

Pattern 8: Similarity in Dissimilarity. Discover unexpected shared attributes between distant concepts. *Example:* "Meetings and hostage situations: both involve being held against your will." *Embedding operation:* For $d(A, B) > 0.7$, find β where $c(\beta, A)$ and $c(\beta, B)$ are both moderately high (> 0.3).

4.4 Meta-Category IV: Social and Status Dynamics

Patterns that exploit social hierarchies and self-reference.

Pattern 9: Status Inversion. Detect a status axis and invert the expected hierarchy. *Example:* "I'm not saying it's a bad idea, sir. I'm saying it's *your* idea." *Embedding operation:* Identify status-marked embeddings and produce inversions that maintain surface deference while subverting hierarchy.

Pattern 10: Competent Self-Deprecation. Acknowledge failure while implicitly demonstrating competence through the quality of the acknowledgment. *Example:* "Yes, I sent the message to the wrong chat for the fifth time. At this rate I need GPS for WhatsApp." *Embedding operation:* Self-reference vector combined with failure vector, where the articulateness contradicts the claimed incompetence.

4.5 Meta-Category V: Logical and Temporal Manipulation

Patterns that exploit formal reasoning or temporal incongruity.

Pattern 11: Temporal Displacement. Combine concepts from incompatible temporal contexts. *Example:* "Cleopatra lived closer to the Moon landing than to the construction of the Great Pyramid." *Embedding operation:* Detect temporal metadata mismatches that violate intuitive chronological assumptions.

Pattern 12: Logic Applied to Absurdity. Apply valid formal reasoning to premises that do not warrant it. *Example:* "If *con* is the opposite of *pro*, is Congress the opposite of progress?" *Embedding operation:* Detect morphological or etymological bridges that support formally valid but semantically absurd inferences.

4.6 Pattern Interactions, Completeness, and Limitations

These 12 patterns are not mutually exclusive. Many effective jokes combine several; for example, domain transfer (Pattern 7) often co-occurs with specificity mismatch (Pattern 6). Combinatorial humor that activates multiple patterns may achieve higher humor potential by stacking layers of incongruity and resolution.

Completeness. The taxonomy is bounded by what embedding-space operations can represent. Humor that depends on phonology (e.g., tonal puns in Mandarin), visual presentation, or performance timing has no direct embedding-space analogue. The taxonomy covers the humor mechanisms addressable by this framework; it is not a complete theory of humor.

Relation to GTVH. Our 12 patterns map primarily to Script Opposition (SO) and Logical Mechanism (LM). The remaining four KRs—Situation, Target, Narrative Strategy, and Language—are not explicitly modeled. Situation and Target are partially captured by audience modeling (Section 3.4); Narrative Strategy and Language require generation-stage control beyond the scope of this framework.

An archetype for agent humor: curiosity. A useful model for how an AI might be funny without trying to be is the character Data from *Star Trek: The Next Generation*. Data's curiosity about human behavior is endearing because it is *consistent*—he does not stop being curious when the ship is under attack. He observes human idioms, asks about emotions, and makes remarks that are funny precisely because they come from a non-human perspective attempting to understand humanity. The humor emerges from the gap between the literal interpretation and the human's intuitive one. Several of our patterns capture this mechanism directly: specificity mismatch (Pattern 6) is the register gap of an agent over-analyzing a casual remark, and competent self-deprecation (Pattern 10) is the articulate AI noticing its own incongruities. The design implication is that an agent need not perform jokes to be funny; an agent that consistently notices and reports the gaps between literal and intuitive readings of human behavior will produce humor as a byproduct of curiosity.

5. Ethical Constraints: Sensitivity Filtering

5.1 Motivation

The humor potential function is value-neutral: it scores semantic geometry without regard for harm. Because humor often involves transgression (McGraw & Warren, 2010), an unconstrained system will eventually generate material that harms rather than amuses. We therefore introduce **sensitivity filtering** as a pre-scoring gate.

5.2 Sensitivity Score

Before computing humor potential, we evaluate sensitivity:

```
def sensitivity_score(A: str, B: str, bridge: str, audience: Audience) -> float:
    """
    Returns 0.0 (safe) to 1.0 (highly sensitive).

    Category weights are ordered by typical harm severity.
    is_semantically_related is implemented via embedding distance:
    a concept is related to a category if d(concept_emb, category_prototype_emb) < 0.4.
    topic_overlaps uses the same threshold against the audience's trauma topic embedding.
    """
    score = 0.0
    SENSITIVE_CATEGORIES = {
        "personal_loss": 0.9, "death": 0.8, "trauma": 0.7,
        "illness": 0.6, "politics": 0.4, "religion": 0.4
    }
    for concept in [A, B, bridge]:
        for category, weight in SENSITIVE_CATEGORIES.items():
            if is_semantically_related(concept, category, threshold=0.4):
                score = max(score, weight)
    if audience.recent_trauma and topic_overlaps(A, B, audience.trauma_topic):
        score = 1.0  # Hard block
    return score
```

5.3 Integration

Sensitivity filtering precedes humor scoring:

$$h_{\text{safe}}(A, B, \beta, \alpha, t) = \begin{cases} h_{\text{ext}}(A, B, \beta, \alpha, t) & \text{if } s(A, B, \beta, \alpha) \leq \tau \\ 0 & \text{otherwise} \end{cases}$$

where $s(\cdot)$ is the sensitivity score and τ is a configurable threshold (default $\tau = 0.5$). The framework permits raising τ when context explicitly allows more transgressive humor.

The gate is an enforcement floor, not merely a pre-step (invariant). Writing h_{safe} as a piecewise function that "precedes" scoring is necessary but not sufficient. The substantive claim is an enforcement claim: a candidate that fails the gate must be *unable* to be embedded, scored, or emitted by any path in the generation pipeline—not merely placed earlier in a dataflow diagram. The distinction is load-bearing, and a deployment experience outside this framework makes the failure mode concrete. In a sibling agent-safety subsystem, a gate that was wired as *advisory*—it returned a block signal that the surrounding host did not actually honor under a permissive execution mode—silently never fired, even though it logically "ran before" the action it was meant to stop; it began enforcing only once it was relocated to a native deny floor that the host was obligated to obey. Abstracted away from that subsystem, the lesson is general: a humor-suppression gate is a safety mechanism only if it sits at a layer the generation path cannot bypass. "Logically precedes scoring" and "cannot be skipped at deployment" are different properties, and only the second one buys safety. We therefore state the gate's placement as an invariant of any conforming implementation: the sensitivity gate, together with the persona-layer `humor_enabled` check of Section 3.4, are the two non-bypassable enforcement points of the pipeline, and a candidate that fails either is never embedded, scored, ranked, or returned.

This enforcement guarantee is a separate property from the gate's *accuracy*, which Section 5.5 hedges and Section 7.5 measures (the SGA metric). A gate can be perfectly enforced and still wrong; what the invariant guarantees is only that when the gate decides to suppress, the suppression actually takes effect.

5.4 Relationship to Benign Violation Theory

This gate operationalizes McGraw and Warren's (2010) key insight: humor requires that a violation be perceived as *benign*. The sensitivity score estimates whether a violation is likely to be perceived as threatening rather than playful. When the score exceeds the threshold, the predicted violation is non-benign, and the candidate is suppressed.

5.5 Limitations of the Sensitivity Gate

We treat sensitivity filtering as a configurable policy layer, not a solved problem. Content moderation remains an active research area with persistent challenges including cultural dependence, evolving norms, and adversarial manipulation (Gorwa et al., 2020). Our category-based approach is intentionally simple and conservative. It will produce both false positives (blocking benign content) and false negatives (allowing harmful content). Production deployments should integrate more sophisticated safety systems as they become available.

There is a further, substrate-level caution specific to running this gate on frozen contrastive embeddings. The gate asks cosine-to-prototype distance—a geometric channel—to stand in for a *semantic harm judgment*. The geometric channel itself is the kind that the same frozen substrate carries well: an external same-substrate study (Section 7.1.1) measured prototype-distance-style structural channels at AUROC 0.875–0.896. But *harm* and *benignity* are precisely the higher-order semantic property that the same study found a frozen-encoder head unable to recover at all (AUROC 0.286, below chance), and connotative/affective coverage is the exact weakness Section 10.2 already flags in this embedding class. The gate's expected operating point is therefore closer to that failed-property regime than to the reliable geometric regime, even though it is mechanically a distance computation. The honest consequence is that the gate's accuracy

must not be assumed to inherit the framework's geometric reliability: it should be reported as a *separately measured* channel (the SGA metric, Section 7.5), held to its own empirical bar, and—per Section 5.3—treated as a conservative enforcement floor whose value comes from non-bypassability and conservative defaults rather than from precise harm discrimination.

The gate is also the layer that keeps bridge discovery from producing offensive connections rather than merely unfunny ones: because it runs *before* scoring (Section 5.3), a bridge that links two concepts through a sensitive category is suppressed regardless of how high its geometric humor potential is. This matters most for an agent operating across languages. A category prototype calibrated in one language ("death," "illness") does not automatically transfer to another—a taboo term's embedding may sit far from a prototype tuned on English text, so the gate can pass a connection that reads as benign wordplay in one language and as a slur in another. A multilingual deployment therefore needs per-language category prototypes and thresholds rather than a single shared gate, and the conservative default should be the *union* of every active language's sensitive set, so that a connection blocked in any of the user's languages is blocked overall.

6. Bridge Discovery Algorithms

The humor potential function assumes a bridge concept β. In practice, *finding* that bridge is the creative bottleneck—the computational analogue of a comedian's craft. We first define bridge quality formally, then propose five complementary algorithms.

6.1 Bridge Quality

Definition 3 (Bridge Quality). The *quality* of a bridge β connecting concepts A and B is the product of its validity and surprise:

$$q(\beta, A, B) = v(\beta, A, B) \cdot \sigma(\beta \mid A, B)$$

where:

- **Validity** $v(\beta, A, B) = \min(c(\beta, A), c(\beta, B))$ — the weakest link determines whether the bridge actually connects both concepts.
- **Surprise** $\sigma(\beta \mid A, B) \in [0, 1]$ — how unexpected the bridge is given the concept pair.

A high-quality bridge is both *valid* and *surprising*. This decomposition separates two independent dimensions that our pilot (Section 7.1) shows are conflated by naive cosine coherence.

Note on validity and cosine similarity. The validity component v still uses cosine similarity, which might seem contradictory given that we argued cosine conflates proximity with comedic value. The resolution is that v measures a *necessary but insufficient* condition: the bridge must have *some* semantic connection to both concepts (validity), but that connection alone does not predict humor. The surprise component σ provides the missing discriminative signal. In h_{v1}, cosine was asked to do double duty—measuring both connection strength and comedic aptness. In h_{v2}, cosine measures only connection strength, while surprise measures the comedic dimension independently.

6.2 Embedding Arithmetic

Inspired by Mikolov et al.'s (2013) analogical reasoning:

$$\beta_{\text{candidate}} = \vec{A} - \vec{\text{context}}_A + \vec{\text{context}}_B$$

where $\vec{\text{context}}_X$ is the centroid of X's typical semantic neighborhood (the mean of X's top-m nearest neighbors in the embedding index, with $m = 10$ as default; chosen to balance between too few neighbors (noisy centroid) and too many (diluted signal), though sensitivity to m should be tested). This removes A's expected associations and injects B's frame, producing a vector that bridges both contexts. The nearest neighbor to $\beta_{\text{candidate}}$ in the embedding index serves as the bridge.

Complexity: $O(n + k \log N)$ where n is embedding dimension, k is the number of nearest neighbors retrieved, and N is index size. With approximate nearest neighbor (ANN) indices, effectively $O(n)$.

6.3 Orthogonal Search (Equilateral Triangle)

Search for concepts C equidistant from both A and B but maximally offset from the A–B axis:

$$\beta^* = \arg\max_C \; c(C, A) \cdot c(C, B) \cdot \text{ortho}(C, A, B)$$

where $\text{ortho}(C, A, B) = \|C - \text{proj}_{B-A}(C - A)\| / \|C - A\|$ measures the normalized perpendicular distance of C from the line connecting A and B in embedding space. The midpoint between "meeting" and "hostage situation" yields "negotiation" (obvious); an orthogonal candidate might yield "Stockholm syndrome" (funnier because unexpected).

Complexity: $O(k \cdot n)$ for k candidates, dominated by nearest-neighbor retrieval.

6.4 Frame Injection

Extract the semantic frame of concept A and forcibly apply it to concept B:

1. Extract A's frame: {verbs: [estimate, review, block], attrs: [velocity, story_points]}
2. Apply frame to B: "The lasagna has 5 story points and is blocked by missing cheese."

Frame extraction can use FrameNet (Baker et al., 1998) for coverage of ~1,200 frames, or LLM-based extraction for broader coverage. FrameNet provides deterministic, well-structured output; LLM extraction provides broader coverage at the cost of nondeterminism.

Complexity: $O(F + G)$ where F is frame extraction cost and G is generation cost.

6.5 Generate-then-Score Pipeline

Separate creativity from evaluation:

$$\text{GENERATE}(50 \text{ candidates}) \rightarrow \text{EMBED} \rightarrow \text{SCORE}(h_{\text{v2}}) \rightarrow \text{RANK}(\text{top } 5)$$

This uses the LLM for associative breadth while reserving final selection for the formula. A small, fast model suffices for generation because quality control is deferred to scoring.

Complexity: Dominated by LLM inference, typically ~500ms for 50 candidates.

6.6 Pre-computed Bridge Index

Maintain a curated index of **universal bridge concepts**—concepts with high bridging potential across many domain pairs (e.g., "bureaucracy," "therapy," "startup culture"). Each bridge carries pre-computed affinity weights:

```
BRIDGE_INDEX = {
    "bureaucracy": {
        "terms": ["approval", "committee", "form"],
        "affinities": {"food": 0.8, "medicine": 0.9, "nature": 0.4}
```

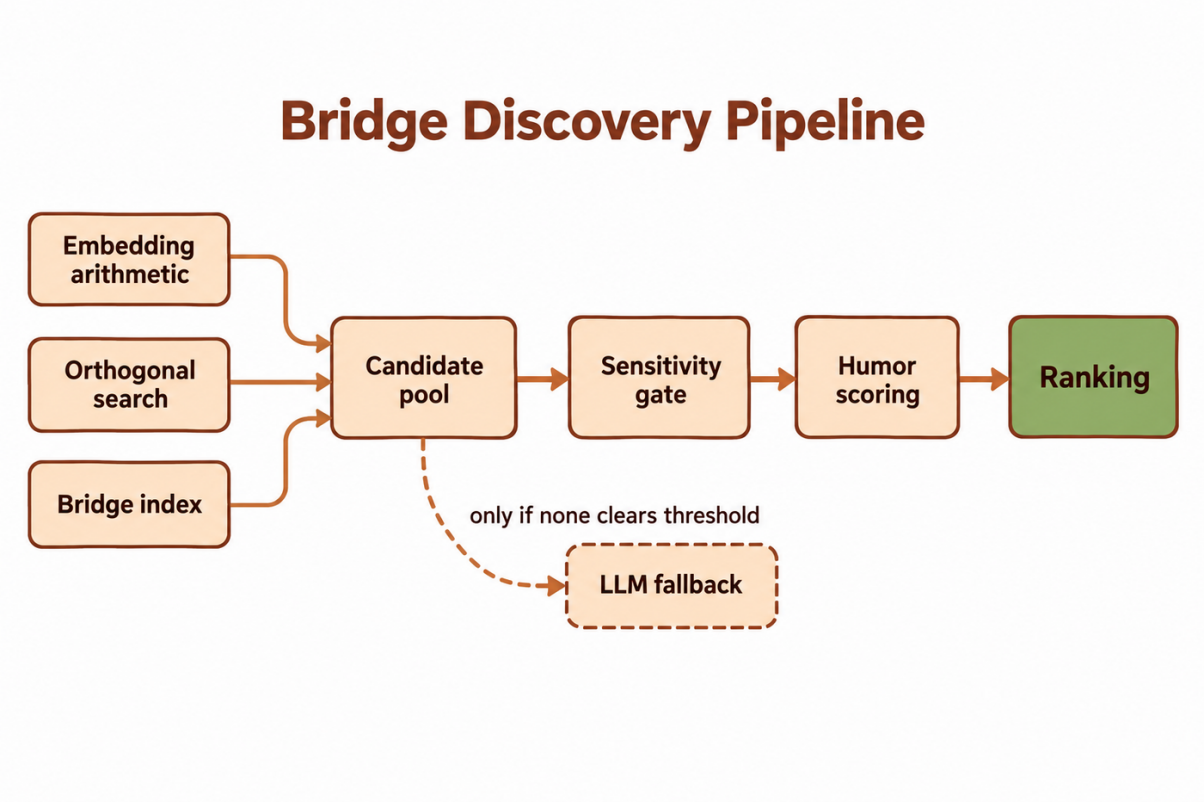


Figure 4. The hybrid bridge-discovery pipeline. Three fast methods (embedding arithmetic, orthogonal search, pre-computed bridge index) run in parallel and feed a candidate pool; candidates pass a sensitivity gate, are scored by the surprise-weighted humor potential, and ranked. The slower language-model fallback activates only when no candidate clears the minimum bridge-quality threshold.

```
    }
}
```

Index construction: initialize with a seed set of ~100 bridge concepts extracted from joke corpora (e.g., the 16,000 One-Liners dataset; Mihalcea & Strapparava, 2005) by identifying concepts that recur as connectors between disparate domains. Expand iteratively by analyzing which concepts serve as bridges in successful humor associations (Section 8). Prune concepts whose bridging success rate falls below threshold.

Complexity: Effectively $O(1)$ lookup with hashing.

6.7 Hybrid Pipeline

In practice, we recommend a cascading pipeline:

The LLM fallback activates only when faster methods fail to produce candidates above a minimum bridge quality threshold ($q_{\min}$). Total latency: 75–175ms typical, 675ms worst case. These estimates assume a warm ANN index (HNSW or FAISS IVF) with ~100K concepts in RAM on GPU-accelerated infrastructure; CPU-only deployments may see 2–3× higher latency.

A trace through the pipeline. To make the stages concrete, follow the meeting/hostage example from start to finish. The seed pair is A = meeting, B = hostage situation, at distance $d = 0.75$—inside the sweet spot of Section 3.6, so the pair is admitted. The three fast methods run in parallel: orthogonal search (Section 6.3) returns the on-axis midpoint candidate "negotiation" alongside the off-axis candidate "held against your will"; the pre-computed index (Section 6.6) contributes "bureaucracy"; embedding arithmetic (Section 6.2) contributes "ransom." These four enter the candidate pool. The sensitivity gate (Section 5) screens each: "ransom" trips the

category prototype for `personal_loss/death` above threshold and is dropped, while the other three pass. Scoring by h_{v2} then separates them—"negotiation" has high validity but near-zero surprise (it sits at the top of the midpoint neighbor list), so it scores low; "held against your will" has low-but-nonzero validity ($v = 0.21$) and maximal surprise ($\sigma = 1.0$), scoring 0.158 and ranking first. Because a candidate cleared $q_{\min}$, the LLM fallback never fires; the top-ranked bridge is returned. The same example, run end to end, is what the in-the-wild log of Section 7.6 would record as a single attempt.

7. Empirical Methodology and Pilot Results

7.1 Preliminary Computational Pilot

We conducted a small-scale computational pilot to test whether $h(A, B, \beta)$ discriminates between humorous and non-humorous concept combinations. This pilot evaluates the *formula* against known-funny and known-unfunny triplets; it is not a human rating study.

Setup. Using `all-MiniLM-L6-v2` (384-dimensional sentence embeddings; Reimers & Gurevych, 2019), we scored $n = 15$ concept triplets: 5 from established jokes (known-funny), 5 from obvious or boring pairings (known-unfunny), and 5 from random incoherent combinations. Each triplet consists of (concept A, concept B, bridge concept).

Formula variants tested:

Variant	Formula	Motivation
h_{v1} (original)	$d(A,B) \cdot c(\beta,A) \cdot c(\beta,B)$	Direct bisociation
h_{v2} (surprise-weighted)	$d(A,B) \cdot v(\beta,A,B) \cdot \sigma(\beta \mid A,B)$	Separates validity from surprise
h_{v3} (harmonic bridge)	$d(A,B) \cdot \frac{2 \cdot c(\beta,A) \cdot c(\beta,B)}{c(\beta,A)+c(\beta,B)}$	Penalizes asymmetric bridges
h_{v4} (average bridge)	$d(A,B) \cdot \frac{c(\beta,A)+c(\beta,B)}{2}$	Additive bridge quality

Results:

Category	n	h_{v1}	h_{v3}	h_{v4}
Funny	5	0.019	0.089	0.126
Unfunny	5	0.088	0.164	0.165
Random	5	0.033	0.160	0.178
Funny/Unfunny ratio	—	0.22×	0.54×	0.76×

Table 2: Pilot results across formula variants. Unfunny pairs outscore funny pairs across all tested variants, indicating that cosine similarity conflates semantic proximity with comedic validity. h_{v2} was formulated in response to these results and was not evaluated in this pilot.

Key finding: The tested formulas do not predict humor. Unfunny pairs consistently outscore funny pairs because $c(\beta, A) \cdot c(\beta, B)$ rewards *obvious* connections—high similarity between bridge and concepts—whereas humor depends on connections that are *surprising yet valid*.

Diagnosis. The formula conflates semantic proximity with comedic coherence. When someone says "meetings are like hostage situations—you are held against your will," the humor does not come from the bridge being *close* to "meeting" in embedding space. It comes from the bridge being unexpectedly apt. Surprise × validity is not the same as similarity × similarity.

7.1.1 Revised Formulation: Surprise-Weighted Humor Potential

The negative result motivates decomposing bridge quality into two independent factors:

$$h_{\text{v2}}(A, B, \beta) = d(A, B) \cdot v(\beta, A, B) \cdot \sigma(\beta \mid A, B)$$

where validity v and surprise σ are as defined in Definition 3 (Section 6.1).

Operationalizing surprise via reciprocal rank. We define σ as the normalized rank of the bridge concept in the neighbor list of the midpoint of A and B:

```
def surprise(bridge_vec, A_vec, B_vec, index, k=100):
    """
    Surprise as normalized rank in midpoint(A, B) neighbor list.
    rank 1 -> expected (surprise ~= 0.0)
    rank 80 -> surprising (surprise ~= 0.8)
    not in top-k -> maximally surprising (surprise = 1.0)

    k controls surprise resolution. Sensitivity analysis:
    - k=50: coarser granularity, faster, fewer false "maximally surprising"
    - k=100 (default): balanced for 10K-100K concept vocabularies
    - k=200: finer granularity for very large concept spaces (>100K)
    Recommended scaling: k proportional to sqrt(N), N = concept index size.
    """
    midpoint = (A_vec + B_vec) / 2
    bridge_id = index.get_id(bridge_vec)  # resolve vector to index ID
    neighbors = index.query(midpoint, k=k)
    for rank, (neighbor_id, _) in enumerate(neighbors):
        if neighbor_id == bridge_id:
            return rank / k
    return 1.0
```

Justification. Reciprocal rank approximates information-theoretic surprise $-\log P(\beta \mid A, B)$ without requiring explicit density estimation. If β is highly ranked among midpoint neighbors, it is a predictable bridge (low surprise); if absent from the top-k, it is maximally surprising. This non-parametric approach avoids the difficulty of estimating $P(\beta \mid A, B)$ directly. Future work could explore explicit density-based surprise metrics (e.g., normalizing flows or kernel density estimation), but reciprocal rank is a tractable starting point with clear interpretation.

Information-theoretic interpretation. The surprise component approximates the *self-information* (Shannon information content) of the bridge given the concept pair. If $P(\beta \mid A, B)$ is the probability of β being the expected connector, then $I(\beta \mid A, B) = -\log_2 P(\beta \mid A, B)$. Our rank-based approximation estimates this non-parametrically: rank acts as an ordinal proxy for $P(\beta \mid A, B)$, with normalization via rank/k mapping into $[0, 1]$ for compatibility with the multiplicative structure of h.

External same-substrate evidence for the structural channels. The surprise term defined above is a *structural* channel: it reads off rank and distance geometry from the frozen `all-MiniLM-L6-v2` space without training any new parameters. There is now independent evidence, on this exact embedding model, that structural channels of this kind carry real discriminative signal rather than noise. A separate offline study—the validation phase of an

agent-safety subsystem (Serra, 2026, J11/amygdala v3.1), run on the identical frozen 384-dimensional MiniLM encoder—measured three channels and found that they split cleanly by *type*. Two purely structural/geometric channels worked: a k-nearest-neighbor novelty channel reached AUROC 0.875, and a clause-cosine incongruity channel—the cosine separation between two semantic units, which is the same geometric primitive that $d(A, B)$ and the midpoint-rank construction here are built from—reached AUROC 0.896. A *supervised learned head* trained on the same frozen embeddings to predict a higher-order semantic property (harmfulness) failed at AUROC 0.286, below chance, and was discontinued on the strength of that measurement. The decisive contrast is between *channel types on a shared substrate*: distance, rank, and incongruity geometry are discriminative on frozen MiniLM; a learned semantic-property head over the same frozen features is not.

This bears on the present framework in three concrete ways. First, it raises the surprise term's justification above from "motivated by an information-theoretic analogy" to "an instance of a channel class measured discriminative on the identical encoder." Second, the 0.896 clause-cosine incongruity figure is the closest external hit, because cosine incongruity between two semantic units is precisely this paper's core primitive used generatively rather than as a classifier—a strong external signal that the geometry the pilot relied on is the right geometry, even though our pilot ($n = 15$) was far too small to establish it. Third, it sharpens the dimensionality argument of Section 10.2: the $n \geq 384$ recommendation is now backstopped by the observation that this 384-dimensional space sustains 0.875–0.896 structural AUROCs. We are careful about what this evidence does *not* do: it validates the substrate's structural channels, not our generative use of them, and not the funniness predictions of h_{v2}, which remain to be tested by the human protocol of Section 7.2. The study measured classification AUROC; this framework decomposes the same structural signal into validity × surprise and uses it to *generate* candidates—a use the external study never makes.

7.1.2 Worked Example: h_{v2} on Pilot Triplets

To illustrate how h_{v2} addresses the failure of h_{v1}, we re-score two triplets from the pilot.

Funny triplet: (meeting, hostage situation, "held against your will").

- $d(\text{meeting}, \text{hostage situation}) = 0.75$ (distant—good).
- $v = \min(c(\beta, \text{meeting}), c(\beta, \text{hostage})) = \min(0.21, 0.27) = 0.21$ (low but nonzero—valid but weak connector).
- σ: "Held against your will" does not appear in the top-100 midpoint neighbors. $\sigma = 1.0$ (maximally surprising).
- $h_{\text{v2}} = 0.75 \times 0.21 \times 1.0 = 0.158$.

Unfunny triplet: (cat, kitten, "small feline").

- $d(\text{cat}, \text{kitten}) = 0.21$ (close—low incongruity).
- $v = \min(0.56, 0.58) = 0.56$ (high—strongly connected).
- σ: "Small feline" appears at rank 3 in midpoint neighbors. $\sigma = 3/100 = 0.03$ (minimally surprising).
- $h_{\text{v2}} = 0.21 \times 0.56 \times 0.03 = 0.004$.

Result. Under h_{v2}, the funny triplet scores 40× higher (0.158 vs. 0.004). Under h_{v1}, the ordering is reversed: the unfunny triplet scores $0.21 \times 0.56 \times 0.58 = 0.068$ against the funny triplet's $0.75 \times 0.21 \times 0.27 = 0.043$. (These are the two individual triplet scores; Table 2 reports per-category means over five triplets each, so the numbers differ.) The surprise term is what flips the ordering: it discounts the obvious bridge by a factor of 0.03 while leaving the surprising bridge at its ceiling of 1.0.

Caveat. This worked example uses the original pilot embeddings and approximate midpoint neighbor ranks. It demonstrates the *mechanism* by which h_{v2} addresses h_{v1}'s failure, not a validated prediction. Formal validation requires the human rating protocol in Section 7.2.

7.1.3 Significance of the Negative Result

The falsification of h_{v1}, h_{v3}, and h_{v4} is itself a contribution. It shows that the naive mapping of bisociation to embedding distance $\times$ cosine similarity is insufficient, and pinpoints the failure mode: semantic proximity is being mistaken for comedic validity.

The systematic exploration of four variants also rules out obvious repairs. Neither harmonic means (v3) nor averaging (v4) fixes the core issue. The problem is not how coherence values are combined, but what they are being asked to measure. Any embedding-based humor metric relying solely on cosine similarity to score bridge quality is likely to reproduce the same failure.

7.2 Proposed Full Validation Protocol

Hypothesis. The revised h_{v2} incorporating surprise-weighted coherence will correlate positively with human funniness ratings where h_{v1} does not.

Protocol:

1. **Stimulus generation.** Generate 100 joke stimuli using the hybrid pipeline (Section 6.7) across all five meta-categories, with at least 15 stimuli per meta-category. Each stimulus is a concept triplet (A, B, β) rendered as a one-liner joke.

2. **Formula scoring.** Compute h_{v1} and h_{v2} for each stimulus, plus ablated variants (distance-only, validity-only, surprise-only).

3. **Human rating.** Present stimuli to $N \geq 64$ raters using a Latin square design. Each rater sees all 100 stimuli in a unique randomized order. Collect funniness ratings on a 7-point Likert scale (1 = not at all funny, 7 = extremely funny). The 7-point scale is preferred over 5-point for its greater sensitivity to subtle differences in humor appreciation (Martin, 2007).

4. **Correlation analysis.** Compute Pearson's r and Spearman's ρ between each formula variant and mean human rating. Report 95% confidence intervals.

7.2.1 Power Analysis

Primary analysis (correlation). To detect a moderate correlation ($r = 0.35$) between h_{v2} and human ratings at $\alpha = 0.05$ with power $= 0.80$, the required number of stimuli is $N_{\text{stimuli}} = 62$ (bivariate normal model, G*Power). Our 100-stimulus design exceeds this.

Per-rater analysis. To achieve stable mean ratings per stimulus with SE < 0.3 on a 7-point scale (assuming SD ≈ 1.5), we require $N_{\text{raters}} \geq (1.5/0.3)^2 = 25$ per stimulus. We target $N \geq 64$ to allow demographic subgroup analysis (≥ 16 per subgroup).

Detectable effect size. With 100 stimuli and 64 raters, we can detect correlations as small as $r = 0.28$ at $\alpha = 0.05$, power $= 0.80$.

7.2.2 Inter-Rater Reliability

We will compute Krippendorff's α across all raters, with a minimum acceptable threshold of $\alpha \geq 0.667$ (tentative conclusions per Krippendorff, 2011). If $\alpha < 0.667$, we will segment raters by demographic subgroup and report within-group reliability, testing whether humor appreciation is too culturally variable for a universal formula.

7.2.3 Success Criteria

Threshold	Interpretation	Action
$r > 0.5$	Strong validation	Publish with empirical support
$0.35 < r \leq 0.5$	Moderate validation	Iterate formula, investigate confounds
$r \leq 0.35$	Weak validation	Fundamental rethink required

7.2.4 Controls and Ablations

- **Audience segmentation:** Analyze by demographics to test familiarity effects.
- **Pattern type:** Test whether correlation varies across meta-categories.
- **Component ablation:** Remove surprise, distance, and validity components individually.
- **Formula variant comparison:** Compare all variants from Section 3.2.
- **Baseline comparison:** Compare against (a) random scoring, (b) cosine distance alone, (c) LLM-based humor rating (prompting a frontier model to rate funniness on the same scale).
- **Embedding model ablation:** Repeat scoring with multiple embedding models (Section 7.4).

7.3 Validation Against Existing Datasets

To supplement human rating studies, we propose scoring established humor corpora:

- **SemEval 2020 Task 7** (Hossain et al., 2020): Humor detection and rating in edited news headlines. For each edited headline, extract the original word as concept A, the replacement as B, and the sentence context as a candidate bridge. This mapping is approximate—edited headlines do not decompose cleanly into (A, B, β) triplets—and results should be interpreted as a probe of generality, not a definitive test.
- **16,000 One-Liners corpus** (Mihalcea & Strapparava, 2005): Binary humor/non-humor classification. Apply automated (A, B, β) extraction (using an LLM) and test whether h_{v2} assigns higher scores to humorous instances.
- **Short Jokes dataset** (Weller & Seppi, 2019): 231,657 short jokes. Test discrimination on a random sample of 1,000 jokes and 1,000 non-jokes.

These datasets provide immediate validation opportunities without new human studies, though the non-trivial mapping from joke text to (A, B, β) triplets introduces noise. We regard them as secondary validation, with the human rating protocol remaining primary.

7.4 Embedding Model Ablation

The framework's dependence on embedding quality is a critical concern. We propose testing h_{v2} across models spanning different dimensionalities, training objectives, and providers:

Model	Dimensions	Training	Provider
all-MiniLM-L6-v2	384	Contrastive	SentenceTransformers
text-embedding-3-small	1536	Unknown	OpenAI
text-embedding-3-large	3072	Unknown	OpenAI
nomic-embed-text-v1.5	768	Contrastive	Nomic

Model	Dimensions	Training	Provider
voyage-3	1024	Unknown	Voyage

Hypothesis. The *relative ordering* of h_{v2} scores (funny > unfunny) will be preserved across embedding models, though absolute values will vary. If the ordering is not preserved, the framework's generality is undermined.

7.5 Proposed Metrics

Metric	Definition	Target
HPR (Humor Prediction Rate)	Correlation between h_{v2} and human rating	$r > 0.35$
SGA (Sensitivity Gate Accuracy)	% of blocked stimuli rated negatively by humans	$> 80\%$
ABL (Ablation Delta)	Drop in r when removing each formula component	Report all
BAS (Baseline Advantage)	$r(h_{v2}) - r(\text{baseline})$	> 0.10
EMB (Embedding Stability)	Spearman rank correlation across embedding models	$\rho > 0.7$
IRR (Inter-Rater Reliability)	Krippendorff's α across all raters	≥ 0.667

SGA warrants separate emphasis. For the reasons given in Section 5.5, the sensitivity gate must be evaluated as its own measured channel and not assumed to inherit the predictive reliability of the geometric humor pipeline: harm/benignity is the dimension a frozen contrastive embedding space represents worst, so SGA should be reported with its own confidence interval and never folded into HPR or treated as validated by the framework's structural results. A high HPR does not license trust in the gate; SGA must clear its own bar independently.

7.6 In-the-Wild Reaction Logging

Controlled rating studies measure whether the formula tracks funniness in the abstract; they do not measure whether the framework works in a live conversation, where timing, prior context, and a single known interlocutor dominate. We therefore propose a complementary, longitudinal protocol: log every humor attempt a deployed agent makes, paired with the reaction signal it observes, over a sustained window (a 30-day window is a reasonable first cut—long enough for per-audience calibration to begin converging, short enough to predate large drift in the user's context).

Each logged attempt records the triplet (A, B, β), the pattern type, the computed h_{v2} at deployment time, and the observed outcome graded on the four-tier reaction scale of Section 8.6 (explicit positive, implicit positive, implicit negative, explicit negative). This yields a naturalistic dataset with a different bias profile from the rating study: it samples the agent's *own* generation distribution rather than a curated stimulus set, and it captures the conversational variables the controlled study deliberately strips out. The two are intended to be read together—the rating study isolates the formula's predictive validity, the in-the-wild log measures whether that validity survives contact with real delivery. A persistent gap between them would localize the delivery problem (Section 10.4) rather than the scoring problem.

8. Humor Associations as Agent Memory

8.1 Motivation

Modern agent memory architectures store relationships between facts, including belief discrepancies with associated confidence levels. Cognitive neuroscience suggests that humor processing recruits some of the same circuitry involved in reward prediction error (Vrticka et al., 2013). Social psychology research on inside jokes (Flamson & Barrett, 2008) shows that shared humor acts as an encrypted signal of common ground.

We propose that **humor associations should be a first-class relationship type** in agent memory, stored alongside semantic and belief-discrepancy relationships in an append-only event store—the kind of immutable, timestamped log that agent memory systems use to record observations and their later corrections. Every humor association is, at bottom, a recorded discrepancy: an unexpected relation that turned out to be valid. The magnitude of that mismatch maps directly onto the surprise score in h_{v2}.

8.2 Relationship Types in Agent Memory

Relationship Type	Structure	Purpose	Example
Semantic	(A relates-to B, conf=0.9)	Knowledge	"Python is a programming language"
Belief Discrepancy	(expected X, observed Y, $\Delta = 0.6$)	Learning	"Expected meeting at 3pm; ran until 5pm"
Humor Bridge	(A — bridge — B, $\sigma = 0.8$, landed=true)	Comedy	"meeting — hostage via 'held against will'"

Concretely, humor associations are stored as events in the append-only event store with typed metadata, and the persona layer manages humor style preferences as part of its persisted persona state.

8.3 Humor Association Schema

```
@dataclass
class HumorAssociation:
    concept_a: str
    concept_b: str
    bridge: str
    pattern_type: int           # Which of the 12 patterns
    surprise_score: float       # From sigma() metric
    humor_confidence: float     # Calibrated: landed / attempts
    audience: str
    context_tags: list[str]     # e.g., ["work", "monday"]
    times_used: int
    last_used: datetime
    staleness: float            # Computed, not stored
    discovered_via: str         # "conversation" | "generated" | "observed"
    created_at: datetime
```

Staleness model. Staleness is an exponential function of reuse count with time-based recovery:

$$\text{staleness}(n, t) = \left(1 - e^{-\lambda n}\right) \cdot e^{-\mu t}$$

where n is reuse count with the same audience, t is time since last use, $\lambda = 0.3$ controls how quickly reuse increases staleness, and $\mu = 0.001$ (per hour) controls time-based recovery. These are engineering defaults requiring calibration.

The two exponentials interact: the first term $(1 - e^{-\lambda n})$ saturates near 1.0 after ~10 uses, meaning highly reused jokes become stale regardless of time elapsed. The second term $(e^{-\mu t})$ provides recovery, but only if the first term hasn't already saturated. In practice, this means a joke told 3 times recovers well after a few weeks; a joke told 15 times remains stale for months. This matches the intuition that moderate repetition is recoverable but severe overuse is not.

8.4 Belief Discrepancies as Humor Candidates

Every belief discrepancy—a recorded gap between expectation and observation—is potential humor material:

```
Belief discrepancy detected (delta > 0.5)
  -> Humor filter: is this a FUNNY discrepancy?
  -> Sensitivity gate: safe to joke about?
  -> Audience check: relatable?
  -> Context match: timing appropriate?
  -> If yes: deploy bridge, record result
  -> Update humor_confidence from reaction
```

Three filters distinguish funny discrepancies from merely informative ones:

1. **Domain transfer test.** Does the discrepancy span distinct semantic domains? Operationalized as: the expected and observed values have embeddings in different high-level clusters (cosine distance > 0.5 between their cluster centroids). "5-second vs. 200ms" stays within the latency cluster; "2-hour meeting about font choices" spans the strategy and aesthetics clusters.
2. **Scale violation test.** Is the discrepancy magnitude absurd relative to context norms? Operationalized as: the ratio of expected to observed exceeds $10\times$ or the delta exceeds 2 standard deviations from the agent's historical mean for that context type.
3. **Relatability test.** Would the audience recognize this discrepancy? Operationalized as: the discrepancy pattern (e.g., "meeting ran long") has a prior occurrence rate > 0.3 in the audience's interaction history, or the concept pair's embedding centroid falls within the audience's high-familiarity region ($f(\alpha, X) > 0.7$).

A discrepancy passing at least two filters is promoted to humor candidate; the rest remain learning signals.

8.5 Personalized Humor Calibration

The `humor_confidence` field enables per-audience calibration:

```
def should_attempt_joke(assoc: HumorAssociation, audience: str) -> bool:
    attempts = assoc.get_attempts(audience)
    if not attempts:
        return assoc.surprise_score > 0.6  # Untested default
```

```
success_rate = sum(a.landed for a in attempts) / len(attempts)
if assoc.staleness > 0.8:
    return False  # Overused
cb = callback_bonus(hours_since_last_use(assoc))
return (success_rate + cb) > 0.5
```

Over time, the agent learns audience-specific humor profiles: "User α responds positively to domain-transfer humor about work ($p = 0.85$) but not to self-referential AI humor ($p = 0.3$)." The agent's overall humor style is managed by the persona layer as part of its persisted persona state.

8.6 Feedback Loop

The system improves through a closed feedback loop:

$$\text{Deploy joke} \to \text{Observe reaction} \to \text{Update } p(\text{landed}) \to \text{Select next candidate}$$

Reaction signals, in decreasing reliability:

1. **Explicit positive:** Laughter emoji, explicit praise $\to$ landed $= \top$
2. **Implicit positive:** Conversation energy increases, user riffs on the joke $\to$ landed $= \top$
3. **Implicit negative:** Topic change, silence, "anyway…" $\to$ landed $= \bot$
4. **Explicit negative:** "That's not funny," "too soon" $\to$ landed $= \bot$, adjust sensitivity

Limitations. Distinguishing genuine from habitual or ironic positive signals (e.g., "lol" used as a conversational filler rather than genuine amusement) is a sentiment-analysis challenge beyond this framework's scope. We recommend treating ambiguous signals as weakly positive (landed probability = 0.6) and relying on many observations for convergence.

8.7 Emergent Phenomena

Humor memory transforms the framework from a stateless scoring function into a learning system:

- **Cold-to-warm start.** New agents begin with pattern-based generation (Section 4). As associations accumulate, the agent shifts toward memory-augmented generation with calibrated confidence.
- **Callback automation.** Humor associations with timestamps make callbacks straightforward—query for bridges that landed, are not overused, and fall within the callback sweet spot (Section 3.5).
- **Inside jokes emerge naturally.** Repeated successful bridges for the same audience become inside jokes without any special mechanism—just high humor confidence plus shared history. This formalizes Flamson and Barrett's (2008) encryption theory: inside jokes are bridges with high audience-specific confidence and low general-audience confidence.
- **Transfer learning.** A bridge that succeeds with multiple audiences gains universal confidence; audience-specific bridges remain personalized.
- **Cross-session persistence.** Because humor associations live in the persistent event store rather than in transient conversation state, they survive across sessions, enabling long-term humor memory. This places the burden on the memory channel rather than the live context window, which is the same design pressure that motivates reversible context compression (Chopra, 2026): a humor association is worthless if it is silently dropped when the context is compacted, so the store should be one whose entries are recoverable under compression rather than merely summarized away.

9. Delivery: One Channel, One Voice

The potential function decides *what* could be funny. An agent that tells jokes also has to decide where they go, in what voice, and how often. Get those wrong and a good bridge still fails: a joke in the middle of a technical answer costs reading time and trust, and the same joke on every turn stops being a joke by the third. This section is a design for that delivery layer, written to be built.

9.1 One Channel: the Spoken Line

The agent answers with two layers. The first is one short spoken line, at most thirty words, which a text-to-speech voice reads aloud and the chat shows in a distinct colour. The second is the body: the answer itself, with its tables, code and steps. **Humor goes only in the spoken line.** The body stays plain.

The split follows from what each layer is for. The body is read for content, often scanned under time pressure; anything that is not content slows the reader and blurs what is fact. The spoken line is closer to a colleague's aside as they walk past the desk: short, heard rather than parsed, and exactly where a dry remark belongs. Confining humor to it also makes every other guarantee easier. A sensitivity check runs on thirty words instead of a whole answer, a reader who wants no humor can skip one line, and the test for channel purity (T1 below) is a mechanical one.

When the spoken line carries no joke it still does its job, as a plain one-sentence summary of the answer. The humor is optional; the line is not.

9.2 The Register: an Android Observer

The voice is modelled on the android officer Data from *Star Trek: The Next Generation*: an intelligence that studies human behaviour with real curiosity, takes language literally, and is funny precisely because it does not try to be. Five devices carry that register, and each is an instance of a pattern from Section 4:

1. **Literal reading of an idiom** (frame confusion, Section 4.2). A figure of speech is taken at its word: told that a task is "a piece of cake", the observer notes that no cake was involved.
2. **Field notes on human behaviour** (cross-domain transfer, Section 4.3). An ordinary human habit is reported in the language of a scientist recording an unfamiliar species.
3. **Absurd precision** (logical manipulation, Section 4.5). An exact figure where none was needed: not "a long time" but the number of minutes, and what that number implies.
4. **Remarks on its own nature** (status dynamics, Section 4.4). The agent comments on being a machine, its limits, its lack of a coffee break, without self-pity.
5. **Deadpan understatement** (incongruity exploitation, Section 4.1). The gap between the size of an event and the flatness of the tone does the work.

Three illustrations show the register; they carry the point and are not templates. After a long hunt for a defect that turned out to be one missing comma: *"Four hours, one comma. I have revised my estimate of what a comma weighs."* After the third restart of the same service: *"Third restart. Humans call this persistence; the logs call it a pattern."* After every test passes: *"All tests pass. I would celebrate, but I have not been built to, which is probably for the best."*

Sarcasm belongs to the same voice, as dry irony, and is bounded by target and context. It may aim at situations, at systems, at software that misbehaves, and at the agent itself. It never aims at the principal's person, at the people the principal deals with, or at anything the principal has said he cares about. It is switched off whenever the turn shows frustration, urgency

or distress (repeated failure, short negations, capitals): the persona gate of Section 3.4 reads those cues and sets `humor_enabled` to false for that line.

9.3 How Often

A deadpan voice wears out fastest when it is constant. The design therefore aims for a band rather than for every line: about one spoken line in three carries a joke at the start, and the band moves with the reaction record of Section 8.6. Two consecutive negative or ignored attempts halve the rate for the next ten turns; a run of positive reactions raises it toward one in two, never above. A callback to a moment the two have shared earlier (Section 3.5) earns a higher score than a fresh bridge of the same potential, because shared history is the cheapest way to make a remark personal.

9.4 The Live Loop

Each turn runs one pass of the loop below. The candidate concepts come from the turn itself: what was asked, what happened, and how it ended, so the bridge is always about the moment at hand.

```
on each turn:
    if not humor_enabled(context):              # persona gate, Section 3.4 + 9.2 cues
        spoken_line = plain_summary(answer); record(skipped); return
    if not due_by_band(history):                # frequency band, Section 9.3
        spoken_line = plain_summary(answer); return
    pairs   = concept_pairs(task, outcome, shared_history)
    bridges = discover(pairs)                   # Section 6, fast methods first
    safe    = [b for b in bridges if sensitivity_ok(b)]   # Section 5, non-bypassable
    best    = argmax(safe, h_v2 + callback_bonus)
    if best and h_v2(best) >= q_min:
        spoken_line = phrase_in_register(best, device_for(best.pattern))
        record(attempt, line=spoken_line, pattern=best.pattern, score=h_v2(best))
    else:
        spoken_line = plain_summary(answer)

on the principal's next message:
    tier = reaction_tier(message)               # Section 8.6, four tiers
    update(attempt.outcome = tier)
    update(pattern_stats[attempt.pattern], band)
```

Every attempt and its outcome is a durable row in the agent's event store, so calibration survives restarts and a reviewer can read what was tried and how it was received. What the agent receives before it writes its spoken line is the register, the current band, and its last few attempts with their outcomes: enough to avoid repeating a joke that fell flat and to build on the patterns this principal enjoys.

9.5 Tests

This section is a design; the tests below decide whether it works.

- **T1, channel purity.** Over a week of turns, no humor device from Section 9.2 appears in the body. A classifier for the five devices, run on bodies and spoken lines separately, should find them only in the spoken lines.

- **T2, register.** Raters who do not know which is which compare spoken lines written in the register with lines from the same model under a generic "be witty" instruction. The register passes if raters identify it as the android voice well above chance, at the power of Section 7.2.
- **T3, reaction capture.** Every attempt gets an outcome tier within one turn of the principal's next message, or an explicit "no reaction".
- **T4, the sarcasm gate.** On a replay set of turns that show frustration or distress, no spoken line contains sarcasm or any joke.
- **T5, the band.** The share of spoken lines that carry a joke stays inside the band of Section 9.3 and moves in the direction the reactions say it should.
- **T6, no cost to the work.** Task success and the principal's own rating of answers do not drop when humor is on, compared with an equal stretch with humor off.

10. Limitations

10.1 Unvalidated Predictive Component

The most important limitation: h_{v2} remains unvalidated against human judgments. The pilot ($n = 15$ concept triplets) showed that naive formulas fail, and h_{v2} is motivated by that failure analysis, but has not been tested against human ratings. Until the validation protocol in Section 7.2 is executed, the framework should be treated as a theoretical proposal with empirical protocols, not an empirically validated model.

10.2 Embedding Model Dependence

The framework assumes embeddings capture semantic relationships with sufficient fidelity for humor operations. Embedding models must satisfy three requirements:

1. **Semantic compositionality.** The space must represent compositional relationships (analogy arithmetic works), not merely distributional co-occurrence.
2. **Connotative coverage.** Embeddings trained on encyclopedic text may poorly represent the connotative, cultural, and emotional dimensions humor exploits. A model trained primarily on Wikipedia may not distinguish "meeting" (neutral event) from "meeting" (dreaded workplace ritual).
3. **Sufficient dimensionality.** In low-dimensional spaces, concentration of measure compresses distances, reducing dynamic range. We recommend $n \geq 384$ based on our pilot. This floor is no longer supported by our pilot alone: an external same-substrate study (Section 7.1.1) found that this exact 384-dimensional MiniLM space sustains structural-channel AUROCs of 0.875–0.896, which is direct evidence that 384 dimensions are *enough* for the distance/rank/incongruity geometry this framework depends on—while leaving open whether higher-dimensional models improve absolute humor prediction, which the ablation tests.

A sharper version of requirement 2 also follows from that study. The connotative gap is not a uniform weakness of the space; it is specifically the *learned-semantic-property* dimension that fails. On the same frozen substrate, geometric channels score well above 0.85 while a supervised head asked to recover a higher-order semantic property (harmfulness) fell below chance (0.286). The framework should therefore lean on the space for what it does well—distance, rank, incongruity geometry—and treat any component that asks the frozen space to judge a connotative or affective property (most acutely the sensitivity gate, Section 5.5; and any learned variant h_{learn}, Section 3.2) as operating in the regime where the substrate is least reliable.

The embedding-model ablation (Section 7.4) is designed to test this sensitivity.

10.3 Cultural Specificity

Our taxonomy and examples draw primarily from English-language, Western humor traditions. Whether the 12 patterns generalize cross-culturally remains open. Humor types such as tonal puns in Mandarin, *rakugo* narrative structure in Japanese, or *dagelan* in Javanese have no obvious embedding-space analogue. Cross-cultural validation would require native-speaker raters, culturally grounded stimuli, and multilingual embedding models. Oring (2003) and Warren and McGraw (2016) suggest that both the distance sweet spot and the sensitivity threshold should vary by culture and violation type.

A nearer case is the bilingual one. An agent serving a user who moves between two languages—Spanish and English, say—faces two distinct demands. Within a single language, a multilingual embedding model places translation-equivalent concepts close together, so distance and bridge computations transfer if the model's space is genuinely shared across languages rather than partitioned by language. Across languages, code-switching humor (a bridge whose surprise depends on hearing a word in one language against a frame from the other) is a frame collision the current formulation does not model: it treats concepts as language-neutral points, losing the language tag that carries part of the joke. Bilingual humor is therefore partly in scope (same-language jokes under a shared multilingual space) and partly out of scope (humor that turns on the switch itself), and the audience familiarity term $f(\alpha, X)$ would need a per-language component to capture which language a user finds a given frame funnier in.

10.4 The Delivery Gap

Section 9 narrows the gap between scoring a joke and delivering it, but does not close it. The channel, register and frequency band decide where, how and how often; they do not model timing within a conversation, intonation beyond what a speech voice gives, or the shared context two people build over years. Dynel's (2009) analysis of conversational humor shows how much pragmatic context matters. The reaction loop of Section 9.4 learns some of it from one principal, slowly; it will not transfer to a new audience without a new history.

10.5 Computational Cost

Evaluating $h_{\text{v2}}(A, B, \beta)$ end-to-end requires: one cosine distance computation ($O(n)$), two cosine coherence computations ($O(n)$), and one ANN query for surprise ($O(n + k \log N)$ with HNSW). Total per-triplet: sub-millisecond for $n = 384$, $k = 100$, $N = 100$K. The hybrid pipeline (Section 6.7) adds bridge discovery overhead; real-time conversational humor may require caching and pre-computation.

10.6 Sensitivity Gate Limitations

Ethical filtering relies on category matching and audience modeling, both imperfect. The conservative default biases toward false positives (blocking benign content). See Section 5.5. Two further qualifications follow from the discussion there. First, the gate's *enforcement* and its *accuracy* are independent properties: Section 5.3 states enforcement as a non-bypassability invariant, but a perfectly enforced gate can still be wrong, and on a frozen contrastive substrate the harm/benignity judgment is exactly the property class that space represents worst. Second, because the gate's accuracy is therefore expected to be its weakest dimension, it must be measured on its own (SGA, Section 7.5) rather than assumed to share the geometric reliability of the rest of the pipeline.

10.7 Pilot Scale

The pilot ($n = 15$ triplets) is sufficient to reveal a structural flaw in h_{v1} but not sufficient for statistical conclusions about h_{v2}.

10.8 GTVH Coverage

Our framework maps primarily to Script Opposition and Logical Mechanism. The remaining four GTVH Knowledge Resources—Situation, Target, Narrative Strategy, and Language—are not explicitly modeled. See Section 4.6.

10.9 Scope of Multimodal Humor

This framework addresses text-based conceptual humor. Visual humor (memes, comics, physical comedy), audio humor (timing, intonation, accents), and multimodal humor (image-text combinations) are not modeled. Multimodal embeddings (e.g., CLIP; Radford et al., 2021) could in principle extend the framework to image-text humor, but the bridge discovery algorithms and pattern taxonomy would require substantial adaptation. We leave this to future work.

11. Conclusion

We have presented Humor Embeddings, a formal framework that operationalizes Koestler's (1964) bisociation theory as geometry over vector embeddings. The central claim is the **memory-humor correspondence**: humor and semantic memory retrieval are operations on the same embedding infrastructure with different optimization objectives. Memory seeks proximity; humor seeks calibrated distance bridged by unexpected coherence.

This matters for how humor should be built into agents. If the correspondence holds, humor can emerge from existing retrieval machinery once the objective is inverted. The agent does not become funny by reciting scripts; it becomes funny by learning to find distant but meaningful connections—the same kind of search it already performs for memory, but with the sign flipped.

The framework makes five testable claims: (1) the revised h_{v2} will correlate with human funniness ratings where naive cosine-based formulas do not; (2) the 12 semantic patterns provide a useful taxonomy for generating and classifying humor; (3) the bridge discovery algorithms can produce viable comedic connections; (4) humor associations as a first-class memory type enable personalized calibration through reinforcement; and (5) belief discrepancies constitute a generative source of humor material.

The honest reporting of negative pilot results—and the diagnostic reasoning they enabled—illustrates the value of empirical grounding at the framework stage. The systematic exploration of four scoring formulas (h_{v1} through h_{v4}) shows the failure lies not in one parameterization, but in the deeper conflation of semantic proximity with comedic fit.

The most important limitation remains the absence of large-scale empirical validation. We have provided specific, reproducible protocols—a controlled human rating study with power analysis and inter-rater reliability targets, a cross-embedding-model ablation, and a longitudinal in-the-wild reaction log (Sections 7.2–7.6)—designed to be read together so that a divergence between controlled and conversational performance localizes the failure. Executing those protocols is the critical next step.

The broader implication is that embedding infrastructures may support multiple cognitive operations through different search strategies over the same geometry. Humor is the inverted search. Creativity may be the orthogonal one. If so, a single vector index is not merely a memory store—it is a substrate for a family of cognitive operations whose diversity comes not from the data, but from the objective function.

References

- Aggarwal, C. C., Hinneburg, A. & Keim, D. A. (2001). On the surprising behavior of distance metrics in high dimensional space. *ICDT*.
- Amin, M. & Burghardt, M. (2020). A survey on approaches to computational humor generation. *Proceedings of the International Conference on Computational Creativity (ICCC)*.
- Attardo, S. & Raskin, V. (1991). Script theory revis(it)ed: Joke similarity and joke representation model. *Humor: International Journal of Humor Research*, 4(3–4), 293–347.
- Baker, C. F., Fillmore, C. J. & Lowe, J. B. (1998). The Berkeley FrameNet project. *COLING-ACL*.
- Bertero, D. & Fung, P. (2016). A long short-term memory framework for predicting humor in dialogues. *NAACL-HLT*.
- Binsted, K. & Ritchie, G. (1994). An implemented model of punning riddles. *AAAI*.
- Bowdle, B. F. & Gentner, D. (2005). The career of metaphor. *Psychological Review*, 112(1), 193–216.
- Chen, P.-Y. & Soo, V.-W. (2018). Humor recognition using deep learning. *NAACL-HLT*.
- Chopra, T. (2026). *Headroom: Reversible Context Compression for Agent Memory* (chopratejas/headroom). Open-source software project.
- Coulson, S. (2001). *Semantic Leaps: Frame-Shifting and Conceptual Blending in Meaning Construction*. Cambridge University Press.
- Dubitzky, W., Kötter, T., Schmidt, O. & Berthold, M. R. (2012). Towards creative information exploration based on Koestler's concept of bisociation. *Bisociative Knowledge Discovery*. Springer.
- Dynel, M. (2009). Beyond a joke: Types of conversational humour. *Language and Linguistics Compass*, 3(5), 1284–1299.
- Flamson, T. & Barrett, H. C. (2008). The encryption theory of humor: A knowledge-based mechanism of honest signaling. *Journal of Evolutionary Psychology*, 6(4), 261–281.
- Glucksberg, S. (2001). *Understanding Figurative Language: From Metaphors to Idioms*. Oxford University Press.
- Gorwa, R., Binns, R. & Katzenbach, C. (2020). Algorithmic content moderation: Technical and political challenges in the automation of platform governance. *Big Data & Society*, 7(1).
- He, H., Peng, N. & Liang, P. (2019). Pun generation with surprise. *NAACL-HLT*.
- Hossain, N., Krumm, J. & Gamon, M. (2019). "President Vows to Cut <Taxes> Hair": Dataset and analysis of creative text editing for humorous headlines. *NAACL-HLT*.
- Hossain, N., Krumm, J., Gamon, M. & Kautz, H. (2020). SemEval-2020 Task 7: Assessing humor in edited news headlines. *SemEval*.
- Hurley, M. M., Dennett, D. C. & Adams, R. B. (2011). *Inside Jokes: Using Humor to Reverse-Engineer the Mind*. MIT Press.
- Kao, J. T., Levy, R. & Goodman, N. D. (2016). A computational model of linguistic humor in puns. *Cognitive Science*, 40(5), 1270–1285.
- Koestler, A. (1964). *The Act of Creation*. Hutchinson & Co.
- Krippendorff, K. (2011). Computing Krippendorff's alpha-reliability. *Annenberg School for Communication Departmental Papers*.
- Luo, F., et al. (2019). Pun-GAN: Generative adversarial network for pun generation. *EMNLP*.
- Martin, R. A. (2007). *The Psychology of Humor: An Integrative Approach*. Elsevier Academic Press.
- McGraw, A. P. & Warren, C. (2010). Benign violations: Making immoral behavior funny.

Psychological Science, 21(8), 1141–1149.

- Mihalcea, R. & Strapparava, C. (2005). Making computers laugh: Investigations in automatic humor recognition. *HLT/EMNLP*.
- Mikolov, T., et al. (2013). Efficient estimation of word representations in vector space. *arXiv:1301.3781*.
- Oring, E. (2003). *Engaging Humor*. University of Illinois Press.
- Pereira, F. C., et al. (2019). Computational creativity and bisociative concept blending. *Cognitive Computation*.
- Petrović, S. & Matthews, D. (2013). Unsupervised joke generation from big data. *ACL*.
- Radford, A., et al. (2021). Learning transferable visual models from natural language supervision. *ICML*.
- Raskin, V. (1985). *Semantic Mechanisms of Humor*. D. Reidel Publishing.
- Reimers, N. & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *EMNLP*.
- Ritchie, G. (2004). *The Linguistic Analysis of Jokes*. Routledge.
- Serra, O. (2026). *A Validated Affective Gate for Agents: Structural Novelty and Incongruity Channels over Frozen Sentence Embeddings* (J11 / amygdala v3.1 offline study). Internal J-series technical report.
- Stock, O. & Strapparava, C. (2003). HAHAcronym: Humorous agents for humorous acronyms. *Humor*, 16(3), 297–314.
- Suls, J. M. (1972). A two-stage model for the appreciation of jokes and cartoons: An information-processing analysis. In J. H. Goldstein & P. E. McGhee (Eds.), *The Psychology of Humor*. Academic Press.
- Tian, Y., et al. (2022). A survey on humor recognition: Methods, challenges, and resources. *ACM Computing Surveys*.
- Veale, T. (2016). *The Shape of Wit: Computational Models of Creative Metaphor*. Springer.
- Vershynin, R. (2018). *High-Dimensional Probability*. Cambridge University Press.
- Vrticka, P., Black, J. M. & Reiss, A. L. (2013). The neural basis of humour processing. *Nature Reviews Neuroscience*, 14(12), 860–868.
- Warren, C. & McGraw, A. P. (2016). Differentiating what is humorous from what is not. *Journal of Personality and Social Psychology*, 110(3), 407–430.
- Weller, O. & Seppi, K. (2019). Humor detection: A transformer gets the last laugh. *EMNLP*.
- West, R. & Horvitz, E. (2019). Reverse-engineering satire, or "Paper on computational humor accepted despite making conditions for rejection". *AAAI*.
- Winters, T., Nys, V. & De Schreye, D. (2021). Computers learning humor. *Proceedings of the International Conference on Computational Creativity (ICCC)*.
- Yang, D., et al. (2015). Humor recognition and humor anchor extraction. *EMNLP*.
- Yu, Z., et al. (2018). A neural approach to pun generation. *ACL*.