How do Language Models Generate Slang: A Systematic Comparison between Human and Machine-Generated Slang Usages

Anonymous ACL submission

Abstract

Slang is a commonly used type of informal language that poses a daunting challenge to NLP systems. Recent advances in large language models (LLMs), however, have made the problem more approachable. While LLM agents are becoming more widely applied to intermediary tasks such as slang detection and slang interpretation, their generalizability and reliability are heavily dependent on whether these models have captured structural knowledge about slang that align well with human attested slang usages. To answer this question, we contribute a systematic comparison between human and machine-generated slang usages. Our evaluative framework focuses on three core aspects: 1) *Characteristics* of the usages that reflect systematic biases in how machines perceive slang, 2) *Creativity* reflected by both lexical coinages and word reuses employed by the slang usages, and 3) *Informativeness* of the slang usages when used as gold-standard examples for model distillation. By comparing human-attested slang usages from the Online Slang Dictionary (OSD) and slang generated by GPT-4o and Llama-3, we find significant biases in how LLMs perceive slang. Our results suggest that while LLMs have captured significant knowledge about the creative aspects of slang, such knowledge does not align with humans sufficiently to enable LLMs for extrapolative tasks such as linguistic analyses.

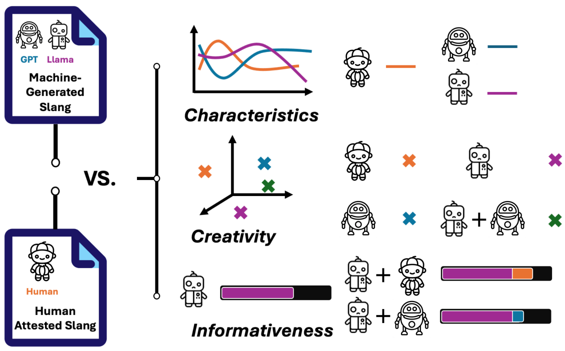

Figure 1: Our evaluative framework considers three core aspects of knowledge: 1) Characteristics, 2) Creativity, and 3) Informativeness by comparing human and machine-generated slang usages. The + sign indicates fine-tuning.

1 Introduction

Slang is a type of informal language that is commonly used in colloquial speech (Sornig, 1981). The use of slang is both creative (Warren, 1992; Eble, 2012) and ephemeral (Eble, 1989), meaning that slang is not only more difficult to comprehend compared to conventional language, but it is also necessary to handle an ever-evolving repertoire of novel slang usages. Such characteristics of slang necessitate natural language processing (NLP) systems that can adapt to unseen slang usages without expansive retraining. While earlier work on NLP for slang often resorts to retrieval-based systems that can hardly generalize (e.g., Pal and Saha, 2013), recent work has been successful in developing systems for the automatic detection (Pei et al., 2019; Liu and Seki, 2021; Sun et al., 2024), generation (Kulkarni and Wang, 2018; Sun et al., 2021), and interpretation (Ni and Wang, 2017; Sun et al., 2022; Mei et al., 2024; Wuraola et al., 2024) of slang that can generalize well toward novel slang usages that have never been seen during training. In particular, large language models (LLMs) have been very effective in many tasks involving slang under both zero-shot and few-shot settings, suggesting that the LLMs have, to some extent, captured structural knowledge about slang that enables generalization.

It is not well understood, however, what underlying structures about slang the LLMs have captured and whether they are comparable to human knowledge. Such insights are valuable in gauging the reliability of downstream applications that require a precise characterization of slang. For example, a model with misaligned knowledge would be systematically biased toward detecting certain types of slang usages. Slang usages generated by such models can also misinform downstream agents that consume the content (e.g., model distillation), as well as linguistic analyses that rely on LLMs for automatic annotation.

We make a first step toward the interpretation of LLMs' internal knowledge about slang by collecting a dataset of machine-generated slang usages from GPT-4o (OpenAI, 2024) and Llama-3 (AI@Meta, 2024) that span a diverse set of controlled conditions. Specifically, we prompt the LLMs to generate novel slang usages that each encompasses 1) A slang term, 2) A sense definition sentence, and 3) A usage context. We perform both controlled generation where the model is provided with existing senses from a slang dictionary and an uncontrolled setting where the model is allowed to generate slang usages attached to any senses. Under each condition, we further constrain the type of word choices that can be made. This includes 1) **Lexical coinage** where the the model is prompted to create a new term, 2) **Word reuse** where the model chooses an existing term in the lexicon, and 3) **Free-form** generation without any restrictions. We collect at least 1,000 valid generations in each setting for a total of 58,197 machine-generated slang usages[1].

Using this dataset, we make a systematic comparison between human-generated slang usages attested by slang dictionaries and machine-generated slang usages. Illustrated in Figure 1, we propose an evaluative framework that makes comparisons along three core aspects: 1) Our framework compares the aggregate usage **characteristics** of the generated slang usages to discern any systematic biases in how machines perceive slang; 2) It measures and compares the **creativity** of slang usages, examining morphological complexity and semantic coherence of coined terms as well as semantic novelty and contextual surprisal for cases of reuse; 3) Model distillation is performed using both human and machine-generated slang usages as examples to measure their **informativeness** toward a diverse set of NLP tasks for slang. Our results show that while LLMs have captured sufficient creative knowledge to generate plausible slang usages, the generated usages still deviate significantly from human-generated usages in certain aspects.

[1]Code and data available at: https://tinyurl.com/msr4r8ez

We make the following contributions in this paper: 1) A dataset of slang usages generated by GPT-4o and Llama-3 that enables studies of machine-generated informal language; 2) An evaluative framework that assesses knowledge alignment between human and machine-generated language use; 3) The first systematic comparison between human and machine-generated slang usages.

2 Related Work

2.1 Knowledge-driven processing of slang

Earlier work relies on building and retrieving from high quality data sources to enable machine processing of slang (Pal and Saha, 2013; Dhuliawala et al., 2016; Gupta et al., 2019). Such approaches require constant updates to obtain new knowledge and thus cannot be efficiently combined with large machine learning models that are expansive to retrain. Meanwhile, encoder models such as BERT (Devlin et al., 2019) perform poorly on slang due to limited scale at the time. To address this, a series of work has been proposed to inject linguistic knowledge about slang into the models to create inductive biases that enable efficient learning. Such an approach has been successfully applied to enable automatic detection (Pei et al., 2019; Liu and Seki, 2021), generation (Kulkarni and Wang, 2018; Sun et al., 2019, 2021), and interpretation (Sun et al., 2022) of slang usages that have not been seen during training.

Most related to our work is Kulkarni and Wang (2018) who built generative neural models of slang word coinage for common types of word formation strategies (e.g., blending) employed in slang word formation identified by prior linguistic research (Mattiello, 2013). Also, Sun et al. (2021) studied cases of word reuse in slang by modeling slang generation as a sense extension phenomenon. In their work, contrastive learning (Baldi and Chauvin, 1993; Bromley et al., 1994; Weinberger and Saul, 2009) was applied to construct a sense embedding space that encapsulates commonly used sense extension patterns (e.g., bad to good) in slang. In our work, we are interested in identifying whether the machine-generated slang usages adhere to such linguistic structures that are informative when modeling both cases of lexical coinage and word reuse.

2.2 Language modeling and slang

As an alternative to knowledge-driven approaches, several studies have explored the possibility of learning knowledge about slang directly from large scale text corpora. Earlier work adopted sequence models for both automatic slang interpretation (Ni and Wang, 2017) and generative word formation (Wibowo et al., 2021). In both cases, the models require a large set of task-specific training data to achieve adequate performance. Recent advances in LLMs have alleviated the need to provide task-specific training data. Sun et al. (2024) evaluated GPT-4 (OpenAI, 2023) on both slang detection and the inference of a slang's demographic origin in zero-shot settings. While task-specific training datasets were still shown to be useful, the zero-shot models show comparable performance with BERT-like models that have been fine-tuned on task-specific data. Wuraola et al. (2024) applied ChatGPT-4 (OpenAI, 2023), Gemini (GoogleAI, 2024a), and Llama-3-8B (AI@Meta, 2024) to slang interpretation and also achieved good performance. Mei et al. (2024) showed that causal inference techniques can be wrapped around LLMs to make further improvements to their predictive accuracies on slang interpretation when compared to traditional prompting methods (e.g., Wei et al., 2022).

Recent progress in this field suggests that LLMs have captured structural knowledge about slang to some extent that enables them to process slang effectively. We extend this line of work by critically examining the prospect of applying LLMs to more complex generative tasks and linguistic analyses involving slang.

3 Data

We first collect sets of slang usages generated by both humans and machines. We use attested slang usages from the Online Slang Dictionary (OSD)[2] as the set of human-generated slang. In OSD, each slang usage consists of:

1. **Lexical term**: A word or phase that denotes slang usage. E.g., "bruddah".
2. **Definition sense**: A definition sentence for the slang sense attached to the lexical term. E.g., "Alternate spelling of brother".
3. **Usage context**: A sentence capturing the context in which the slang is being used. E.g., "Safe, my *bruddah*".

We obtain 9,115 slang usage entries from OSD by sampling one usage from each unique term.

[2]http://onlineslangdictionary.com/

For machine-generated slang, we use GPT-4o and Llama-3-8B to each generate a set of slang usages. We consider two generation settings. First, we perform **controlled** generation where each generation prompt is conditioned on an existing definition from OSD. Here, The model is asked to assign a slang term that would express the given human-defined meaning along with an example usage context. This setup focuses on making word choices grounded in existing concepts. We also perform **uncontrolled** generation where the model is able to express concepts outside ones attested in the slang dictionary. In this case, the generated usages rely solely on the model's intrinsic knowledge about slang obtained during pre-training. Under each setting, we further control for the type of word choice made by the model. Under the *Coinage* condition, the model is prompted to generate a novel term. Conversely, under the *Reuse* condition, the model is prompted to assign an existing term. Finally, we include a *Free-form* condition where the model can pick either types. We ensure compliance by checking the model generated terms against the English Wiktionary (Ylonen, 2022)[3]. We filter out all model outputs that do not conform to the instructions. Table 1 summarizes all control conditions used for generation and the corresponding data partitions. We will refer to the partitions outlined in Table 1 throughout the paper.

Under the uncontrolled condition, we generate slang usages iteratively until we reach 1,000 usage entries. To enhance the diversity of the generated slang usages, we first implemented the method proposed by Chen et al. (2024) which introduces prompt-level randomness by prompting with a randomly generated number attached to each instance of generation. However, it yielded suboptimal results compared to the baseline (see Appendix A for an ablation). As a result, we adopted a simple sampling configuration with a temperature of 1.2 and top-p of 0.95. We use default values for all other hyper-parameters. This best-performing configuration yielded 765 unique compliant entries out of the first 1,000 generations.

To remove duplicates in generation, we ensure that each generated item must contain either a unique slang term or sense definition. We allow duplicate terms only if their associated senses are semantically distinct, defined as having a cosine similarity below 0.8 between their corresponding *all-*

[3]See details in Appendix F.

Setting	Free-form	Reuse	Coinage
Uncontrolled	U-F	U-R	U-C
Controlled	C-F	C-R	C-C

Table 1: Data partitions for slang usage generation under all possible experimental conditions.

MiniLM-L6-v2 Sentence-BERT (SBERT, Reimers and Gurevych, 2019) embeddings.

To avoid generating duplicates over overlapping senses in the controlled generation setting, we apply clustering to all definition sense embeddings. Specifically, we encode all sense definition sentences from OSD using SBERT and apply clustering with DBSCAN (Hahsler et al., 2019) using default hyperparameters. Out of the 9,115 usage entries from OSD, the DBSCAN clustering procedure yielded 7,890 distinct word sense clusters.

For each sense cluster $\mathcal{C} = \{s_1, s_2, \ldots, s_n\}$, we choose the most frequent sense definition sentence s^* to be used in the prompt. If necessary, we break ties by random sampling. Given s^*, we prompt the LLM to generate a list of terms $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ and contexts $\mathcal{C} = \{c_1, c_2, \ldots, c_n\}$ such that each term t_i and context c_i defines a slang usage with definition s^*. We perform the generation iteratively and reject any duplicated terms that have already been generated until we have n generated usages. The prompts used for data collection are included in Appendix D. The algorithms used for the prompting and filtering processes are in Appendix F. Examples from both OSD and our generated datasets can be found in Appendix H.

4 Experiments

4.1 Characteristics

We first compare and examine the aggregate characteristics of human and machine-generated slang usages with respect to usage types, word formation patterns, and expressed topics. Figure 2 shows the distribution of the human and machine-generated slang across two usage types: lexical coinage and word reuse. We observe clear distinctions between slang usages from OSD and the LLMs. While OSD exhibits a more balanced distribution across the two usage types, both GPT-4o and Llama-3 display a strong inclination toward producing coinages. In the uncontrolled case, GPT-4o only produced 3 coinages out of 1,000 generations. Interestingly, prompting GPT-4o using slang senses from human attested usages (i.e., controlled case) significantly reduces the imbalance. The proportion of reuse

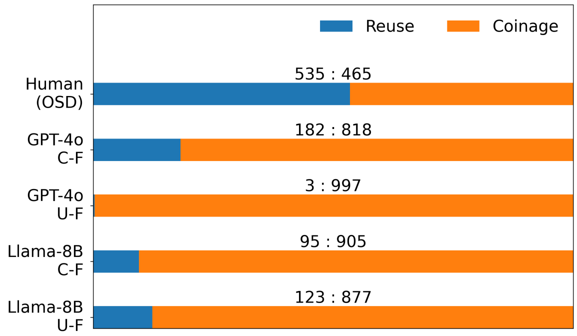

Figure 2: Reuse–Coinage proportion of human (OSD) vs machine-generated slang.

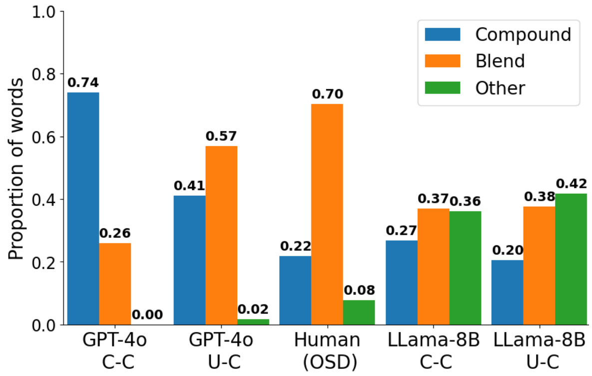

Figure 3: Distribution of word formation processes used in both human and machine-generated slang.

cases, however, is still much lower compared to human-generated OSD. The machine-generated slang shows that LLMs' perception of slang is heavily biased toward the use of novel terms.

For cases of coinage, we also examine the word formation processes that are employed. In the case of human-generated usages, a set of common word formation processes have been identified in the literature (Mattiello, 2013). One of the most prominent processes is blending, where parts of two lexical items are combined to create a new term (e.g., *lambortini*). Related to blending is the process of compounding (Lehrer, 1970; Brinton and Traugott, 2005) in which two words are combined verbatim (e.g., *backwash*). Here, we approximate the proportion of the coinage cases that employs either compounding or blending using lexical decomposition (see detailed algorithm in Appendix G). First, we use Morfessor (Smit et al., 2014) to perform morphological segmentation on each slang term and only consider words with more than two morphological segments in this experiment. We then query each of the two segments in Wiktionary. If exact matches can be found for all segments, we label the term as a compound. Otherwise, if a segment

Topic	OSD Topic Words	GPT-4o Topic Words
Topic 1	find, people, area	movement, energetic, digital
Topic 2	acronym, sexual, p**is	energy, excitement, enthusiasm
Topic 3	person, money, f***	playful, laughter, fleeting
Topic 4	sex, sh**, spell	unexpected, surprise, reaction
Topic 5	female, term, attractive	quick, smile, attention

Table 2: Representative topic words from both OSD and GPT-4o sense definitions. Full results are in Appendix I.

can be found as a prefix of a word in Wiktionary and the next segment can be found as a suffix then the word is labeled as a blend. All other types of coinages are labeled as 'Other'. We apply the labeling routine to all coinage cases in OSD and both the controlled and uncontrolled machine-generated slang usages generated under the coinage condition. Figure 3 shows the proportion of coinage cases that have been labeled into each word formation categories. We observe similar distributions from the uncontrolled GPT-4o generations and human data with GPT-4o biased toward more compounds than blends. Interestingly, when we control GPT-4o using definitions from OSD, the model shows an even stronger bias toward coining compound words. Llama-generated usages, on the other hand, show much less preference toward either compounding or blending compared to both human and GPT-generated slang. The distinction in these distributions shows that individual LLMs obtain its own perception of slang that is not necessarily well-aligned with human knowledge.

Finally, we examine topical preferences in the slang usages by applying LDA topic modeling (Blei et al., 2003) on 1,000 definition sentences (filtering out all stop words) each from both OSD and the U-F set from both LLMs. We use Gensim (Řehůřek and Sojka, 2010) to extract the 20 most representative words under 5 topics. Table 2 shows example words from each topic. Consistent with previous findings (Labov, 1972, 2006), our results show a strong preference toward taboo topics such as sex and profanity in real slang from humans. Meanwhile, machine-generated slang shows a notable preference toward more positive but less concrete concepts. One hypothesis is the influence of alignment techniques (e.g., RLHF; Ouyang et al., 2022) prevents the models from producing outputs involving potentially offensive or controversial content and instead steers them toward neutral or positive expressions. Our results reaffirm that human slang are created to reflect cultural dynamics while suggesting that LLMs' generations merely capture the creative aspect of the generative process.

Source	Mean	Std
Human (OSD)	2.032	0.841
GPT-4o C-C	2.442	0.797
GPT-4o U-C	2.634	0.655
Llama-3-8B-INT	1.698	0.741
Llama-8B + GPT4o C-C	1.985	0.807
Llama-8B + GPT4o U-C	2.038	0.748
Llama-8B + OSD-C	1.811	0.794

Table 3: Morphological complexity scores for coined terms measured by the number of segments in their respective Morfessor decompositions.

Source	Mean	Std	IQR	Kurtosis
Human (OSD)	1.286	0.058	0.076	0.185
GPT-4o C-C	1.277	0.055	0.069	0.275
GPT-4o U-C	1.250	0.058	0.076	0.136
Llama-3-8B-INT	1.290	0.061	0.085	1.180
Llama-8B + GPT-4o C-C	1.291	0.059	0.075	0.270
Llama-8B + GPT-4o U-C	1.274	0.055	0.074	0.219
Llama-8B + OSD-C	1.308	0.049	0.065	1.957

Table 4: Morphological coherence scores for compound words across all sources. Lower values indicate better semantic alignment between the slang senses and the coined terms.

4.2 Creativity

4.2.1 Creativity in Coinage

We evaluate the morphological creativity of coined slang terms through two key aspects: *morphological complexity* and *morphological coherence*. We define morphological complexity as the average number of morphological segments a coined term has. Here, higher complexity reflect more elaborate word composition strategies being employed. We also measure the morphological coherence of each coined compound by comparing semantic representations of the slang sense with senses corresponding to each constituent word. Here, better coherence suggests that the coined term is more semantically grounded with respect to its morphological structure. While morphological segmentation reflects surface-level complexity, coherence measures whether the meaning of the coined word is semantically consistent with its constituent morphological segments, thus reflecting a more nuanced level of creativity.

Morphological Complexity. We use Morfessor to decompose all slang terms that are not found in Wiktionary. Table 3 presents the average number of morphological segments per coined term in both OSD and machine-generated coinages. We find that GPT-coined terms are much more complex compared to coinages from OSD, for both the controlled and uncontrolled conditions, especially in the uncontrolled case where the model also shows significantly lower variability in complexity. Meanwhile, The Llama model produces much simpler constructions compared to both OSD and GPT-4o, suggesting that different LLMs have varied preferences toward lexical creativity in slang.

Morphological Coherence. We compute the morphological coherence score for all compounds by taking the average Euclidean distance between the SBERT embedding of the coined term's slang definition and the embeddings of sense definitions corresponding to the term's constituent words. Table 4 reports the results. Among the models, GPT-4o under the uncontrolled setting achieved the lowest mean score, indicating that its coined terms are not only morphologically complex but also more semantically coherent. In contrast, coherence degrades notably when GPT-4o is conditioned on human attested senses (C-C), reflecting a reduction in semantic consistency. The results indicate that GPT-4o prefers more semantically coherent word choices when coining new words while humans are more playful in their word choices.

4.2.2 Creativity in Reuse

Aside from lexical coinage, slang also employs flexible reuse of existing words in creative ways (Warren, 1992). We measure creativity in two distinct aspects. First we measure the *novelty* of the slang sense extension encompassed by the word choice in a reuse. Motivated by computational models of slang reuse (Sun et al., 2019), we measure novelty by computing the semantic distance between the intended sense S of the slang term with the prototypical sense representation (Rosch, 1975) of the reused term t:

$$\text{Novelty}(t, S) = \|E(S) - \frac{1}{|\mathcal{C}_t|}\sum_{i=1}^{|\mathcal{C}_t|} E(\mathcal{C}_{t_i})\|_2$$

Here, $\mathcal{C}_t$ denotes the set of existing senses of the term t in a conventional dictionary and $E(\cdot)$ is an embedding function. The prototypical sense representation encapsulate the aggregate meaning of t and thus the difference between it and the slang sense representation reflects novelty of the sense extension.

We compute an alternative measure of creativity in reuse inspired by the Cooperative Principle of Grice (1975) and Principle of Relevance proposed by Sperber and Wilson (1986). Under these frameworks, the lack of relevance in the use of a lexical item indicates the speaker's intention to invoke a novel or enriched meaning. In extension, we infer the creativity of a slang reuse by computing its *surprisal* in context. We operationalize surprisal by computing the average negative log-likelihood score assigned to all constituent tokens of a slang term by the LLM conditioned on the proceeding context. Higher surprisal indicates that the model finds the term less predictable in the given context. The surprisal score thus measures the creativity of the slang usage with respect to the usage context.

Model	Mean	Std	IQR	Kurtosis
Human (OSD)	1.141	0.209	0.220	4.591
GPT-4o C-R	1.231	0.127	0.130	6.870
GPT-4o U-R	1.226	0.107	0.118	4.658
Llama-3-8B-INT	1.222	0.124	0.144	3.190
Llama-8B + GPT4o C-R	1.257	0.108	0.123	4.095
Llama-8B + GPT4o U-R	1.252	0.104	0.121	2.790
Llama-8B + OSD-R	1.257	0.106	0.127	5.691

Table 5: Summary of novelty statistics across all cases of word reuse.

Source	Mean	Std	IQR	Kurtosis
Human (OSD)	28.73	13.06	17.59	0.73
GPT-4o C-R	27.54	11.44	16.06	1.73
GPT-4o U-R	26.33	10.23	15.75	-0.30
Llama-3-8B-INT	30.62	12.71	19.28	0.11
Llama-8B + GPT4o C-R	29.83	12.18	18.00	0.14
Llama-8B + GPT4o U-R	28.42	11.96	18.86	0.30
Llama-8B + OSD-R	27.70	12.71	19.91	0.73

Table 6: Summary of surprisal statistics across all cases of word reuse. Lower surprisal score means better coherence w.r.t. usage contexts.

Novelty. Table 5 shows the mean novelty scores computed over sets of slang usages. Both GPT-4o and Llama achieve high mean novelty scores across all settings, suggesting that LLMs can consistently generate more semantically divergent slang usages. Notably, human-generated OSD usages exhibits the lowest mean novelty but significantly higher dispersion indicated by high standard deviation and IQR. Our results suggest that human speakers generate slang usages in a much wider creative spectrum with a relatively loosely defined level of creativity attached to the use of slang. Meanwhile, slang reuse generated by the LLMs tend to cluster around a specific level of creativity that's more creative than the average human-attested usage.

Surprisal. We measure the surprisal scores using Gemma-2-9b-Intruct (GoogleAI, 2024b) as a judge (Zheng et al., 2023), which is not a member of neither the GPT or Llama model family to ensure objectivity. Table 6 shows the surprisal values. We find the surprisal values to be high across the board, indicating that machine generated slang shows nuanced control over contextual surprisal similar to those found in human usages. We find that GPT-4o in the controlled setting produced slang usages that are slightly more coherent to the context compared to humans, while the uncontrolled Llama model generates less contextually coherent usages despite having a similar level of novelty.

Overall, our results suggest that while LLMs are capable of generating creative slang usages similar to how humans do, the larger GPT-4o model tends to prefer generations that are more creative in certain aspects (i.e., morphological complexity, semantic novelty) than others (i.e., morphological coherence, contextual surprisal).

4.3 Informativeness

In this section, we ask the question of whether machine-generated slang shares a similar level of informativeness when used as examples. If an LLM truly captures the nuanced generative structure behind slang usages, we would expect the machine-generated slang usages to be as informative as human generated samples. To do this, we compare the informativeness of GPT-4o generated samples with human-generated slang under a distillation experiment (Hinton et al., 2015). Specifically, we use Llama-3-8B-Instruct as the student model and finetune it using slang usages from each source.

To ensure valid comparison and evaluation, we construct a balanced sample of slang usages with 1,000 training examples in each partition. We begin the sampling procedure by first splitting the entire OSD dataset into an 80-20 split for training and testing respectively. From the OSD training set, we randomly sample 1,000 examples for training. We also ensure that the 1,000 examples correspond to different sense clusters to maintain semantic diversity. For the GPT-4o generated partitions with controlled generation, we sample usages generated from examples in the OSD training set to avoid data contamination. We obtain three samples from OSD with 1,000 examples in each, corresponding to free-form (OSD-F), coinage (OSD-C) and reuse (OSD-R) respectively. We also obtain 1,000 examples for each of the machine-generated partitions outlined in Table 1. An additional 80-20 split is then applied to each 1,000-example set to create the final training and validation partitions used for model fine-tuning using LoRA (Hu et al., 2022). Detailed experimental setup and prompts can be found in Appendix B and E.4.

4.3.1 Informing creativity

We first finetune the Llama-8B model on either cases of coinage or reuse from both OSD and GPT-4o generated usages to examine the effect of fine-tuning on the model's creativity. Using the fine-tuned models, we perform uncontrolled generation of 1,000 samples from each model and repeat the experiments in Section 4.2. Table 3 and 4 show the results for coinages and Table 5 and 6 show the results for reuses.

we observe a mild signal for the transfer of coinage creativity: When Llama-8B is fine-tuned on coinages from either GPT-4o or OSD, we observe a significant increase in morphological complexity, more so when finetuned on GPT-generated slang. It shows that learning on more morphologically complex examples generated by GPT-4o does steer the student model toward generating more morphologically complex slang terms. Meanwhile, fine-tuning the model on OSD-generated entries makes the generations less coherent while fine-tuning on GPT-4o's uncontrolled generations makes them more coherent, once again showing that the fine-tuned model is being steered toward mimicking the preference of the teacher model.

When fine-tuned on reuse cases, we observe that while the Llama-generated slang attains a comparable increase in semantic novelty, the level of surprisal decreases across all cases, particularly when the model is fine-tuned on OSD slang. Interestingly, although slang usages from GPT-4o were more coherent w.r.t. the usage contexts, such knowledge does not transfer well in the fine-tuning process.

Overall, our results show that both human and machine-generated slang can provide informative information in some dimensions of creativity by steering the smaller model toward mimicking the larger model's preferences. It is important to note here that using large models such as GPT-4o as a teacher will propagate many of its biases into the smaller student model and caution should be exercised in scenarios where we want the models to faithfully represent human knowledge.

4.3.2 Informing knowledge

We now examine the informativeness of slang usages for downstream tasks that require structural understanding of slang. We evaluate the performance of the vanilla and fine-tuned Llama models

Model	Task 1 (Acc)	Task 2 (Acc)	Task 3 (Sim)
[OSD]			
GPT-4o	0.959 ± 0.003	0.989 ± 0.004	0.520 ± 0.001
Llama-3-8B-INT	0.891 ± 0.001	0.910 ± 0.000	0.471 ± 0.001
Llama-8B + GPT-4o C-F	0.886 ± 0.000	0.913 ± 0.001	0.494 ± 0.000
Llama-8B + GPT-4o U-F	0.889 ± 0.001	0.914 ± 0.000	0.486 ± 0.000
Llama-8B + OSD-F	0.882 ± 0.000	0.914 ± 0.000	0.500 ± 0.000
[OpenSubtitles-Slang]			
GPT-4o	0.965 ± 0.001	0.913 ± 0.004	0.501 ± 0.001
Llama-3-8B-INT	0.928 ± 0.000	0.844 ± 0.000	0.464 ± 0.000
Llama-8B + GPT-4o C-F	0.924 ± 0.000	0.838 ± 0.000	0.481 ± 0.000
Llama-8B + GPT-4o U-F	0.922 ± 0.000	0.852 ± 0.000	0.475 ± 0.000
Llama-8B + OSD-F	0.926 ± 0.000	0.828 ± 0.000	0.487 ± 0.001

Table 7: Performance of models across three evaluation tasks over 10 runs. Task 1 (Generation) and 2 (Interpretation) are measured by accuracy and Task 3 (Free-form interpretation) is measured by semantic similarity using SBERT.

on both slang generation and slang interpretation:

Task 1 - Generation (*Cloze + Definition → Word*) evaluates the model's ability to infer a slang term given a masked usage sentence (i.e., the slang term is masked out) and its corresponding definition. The model is given four choices and asked to pick the correct word choice.

Task 2 - Interpretation (*Word + Usage → Definition*) presents a slang word in a usage sentence and asks the model to select the correct definition among four choices.

Task 3 - Free-form interpretation (*Word + Usage → Definition (Generation)*) employs a similar setup as the interpretation task but here the model needs to generate a definition sentence for the meaning of the slang. The generated definitions are evaluated by their semantic similarity to the ground-truth definition sentence using SBERT.

We evaluate both the off-the-shelf GPT-4o and Llama-3-8B-Instruct model with fine-tuned Llama models on all three tasks. We construct evaluation datasets using both OSD and OpenSub-Slang, a benchmark dataset on slang curated by Sun et al. (2024) using the OpenSubtitles corpus (Lison and Tiedemann, 2016). Specifically, OSD evaluates informativeness of the generated usages in an intrinsic setting where the evaluation data shares the same data distribution as the entries used to both fine-tune Llama and control GPT-4o's generation. OpenSub-Slang, on the other hand, provides an extrinsic evaluation. See Appendix C for detailed experiment setup.

Table 7 summarizes the results. GPT-4o consistently outperforms Llama models across all tasks, reflecting its robust knowledge on slang. The vanilla Llama model lags behind GPT-4o by a significant margin. We observe that finetuning on both OSD and GPT-4o generated slang yields no or minimal performance gain on tasks 1 and 2, suggesting that the student models are not able to effectively leverage the knowledge to make predictions. On Task 3, finetuning on both data sources improves the quality of the generated definitions, arguably because the model is better guided toward writing definition sentences that better conform with the dictionary style. In this case, we find human-generated OSD to be more informative compared to slang entries generated by GPT-4o. Overall, while smaller Llama models can sometimes benefit from the transfer of knowledge, the degree of improvement is task-sensitive and often constrained.

5 Conclusion

We have presented the first systematic comparison between human and machine-generated slang usages. Our results suggest that while LLMs such as GPT-4o achieve strong results on a wide range of evaluative tasks involving slang, structural knowledge about slang encoded in these models show notable distinctions when compared to real usages from humans. Specifically, LLMs show strong preferences toward certain characteristics and creative qualities, and such preferences can affect how the generated usages inform their users. Our findings suggest that although LLMs are capable of processing slang in ways that reflect many aspects of human knowledge, they have not yet fully captured nuanced structures in human slang usage.

Limitations

Due to computational budget constraints, we limit our evaluation of large language models to only Llama-8B-INT and GPT-4o. Although this combination captures a representative sample of both open and black-box commercial models, they do not fully capture the diverse landscape of contemporary LLMs. Ideally, we would like to expand our analysis to include a broader range of commercial systems to better understand how different models behave, as well as running large variants of open models such as Llama-70B.

The scope of our study is also confined to English slang where both the human and machine-generated slang entries are only in English. Slang is inherently a cultural and multilingual phenomenon, thus evaluating how models handle slang in other languages/dialects remains an important direction for future work. Addressing these gaps would help assess whether the interpretative results we represented can be generalized across linguistic and cultural boundaries.

Ethics Statement

We acknowledge that many slang usages collected from dictionaries express taboo concepts. In the main text, we mask out certain example words that are deemed inappropriate and present the full results in the Appendix. All slang usages shown in the examples were taken verbatim from the original data source and do not reflect opinions of the authors and their affiliated organizations. Discretion is advised when viewing the examples in the Appendix and using the collected datasets.

We have been granted written permission from the author of The Online Slang Dictionary to use it for personal research purposes.

We used AI assistants to expedite the coding process. All code snippets produced by AI assistants were verified by the first author before they were incorporated. For writing, we only used AI assistants to check grammar.

References

AI@Meta. 2024. The llama 3 herd of models. *arXiv*.

Pierre Baldi and Yves Chauvin. 1993. Neural networks for fingerprint recognition. *Neural Computation*, 5(3):402–418.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3(null):993–1022.

Laurel J. Brinton and Elizabeth Closs Traugott. 2005. *Lexicalization and Language Change*. Research Surveys in Linguistics. Cambridge University Press.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1994. Signature verification using a "siamese" time delay neural network. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pages 737–744. Morgan-Kaufmann.

Jiuhai Chen, Rifaa Qadri, Yuxin Wen, Neel Jain, John Kirchenbauer, Tianyi Zhou, and Tom Goldstein. 2024. Genqa: Generating millions of instructions from a handful of prompts. *arXiv*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shehzaad Dhuliawala, Diptesh Kanojia, and Pushpak Bhattacharyya. 2016. SlangNet: A WordNet like resource for English slang. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4329–4332, Portorož, Slovenia. European Language Resources Association (ELRA).

Connie C. Eble. 1989. The ephemerality of American college slang. In *The Fifteenth Lacus Forum*, 15, pages 457–469.

Connie C. Eble. 2012. *Slang & Sociability: In-group Language among College Students*. University of North Carolina Press, Chapel Hill, NC.

GoogleAI. 2024a. Gemini: A family of highly capable multimodal models. *arXiv*.

GoogleAI. 2024b. Gemma 2: Improving open language models at a practical size. *arXiv*.

H. Paul Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and Semantics, Volume 3: Speech Acts*, pages 41–58. Academic Press.

Anshita Gupta, Sanya Bathla Taneja, Garima Malik, Sonakshi Vij, Devendra K. Tayal, and Amita Jain. 2019. SLANGZY: a fuzzy logic-based algorithm for english slang meaning selection. *Progress in Artificial Intelligence*, 8(1):111–121.

Michael Hahsler, Matthew Piekenbrock, and Derek Doran. 2019. dbscan: Fast density-based clustering with R. *Journal of Statistical Software*, 91(1):1–30.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Vivek Kulkarni and William Yang Wang. 2018. Simple models for word formation in slang. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1424–1434, New Orleans, Louisiana. Association for Computational Linguistics.

William Labov. 1972. *Language in the inner city: Studies in the Black English vernacular*. University of Pennsylvania Press.

William Labov. 2006. *The social stratification of English in New York City*. Cambridge University Press.

Adrienne Lehrer. 1970. Notes on lexical gaps. *Journal of Linguistics*, 6(2):257–261.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

Yihong Liu and Yohei Seki. 2021. Joint model using character and word embeddings for detecting internet slang words. In *Towards Open and Trustworthy Digital Societies: 23rd International Conference on Asia-Pacific Digital Libraries, ICADL 2021, Virtual Event, December 1–3, 2021, Proceedings*, page 18–33, Berlin, Heidelberg. Springer-Verlag.

Elisa Mattiello. 2013. *Extra-grammatical Morphology in English: Abbreviations, Blends, Reduplicatives and Related Phenomena*. De Gruyter Mouton, Berlin, Boston.

Lingrui Mei, Shenghua Liu, Yiwei Wang, Baolong Bi, and Xueqi Cheng. 2024. SLANG: New concept comprehension of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12558–12575, Miami, Florida, USA. Association for Computational Linguistics.

Ke Ni and William Yang Wang. 2017. Learning to explain non-standard English words and phrases. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 413–417, Taipei, Taiwan. Asian Federation of Natural Language Processing.

OpenAI. 2023. GPT-4 technical report. *arXiv*.

OpenAI. 2024. GPT-4o system card. *arXiv*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Alok Ranjan Pal and Diganta Saha. 2013. Detection of slang words in e-data using semi-supervised learning. *International Journal of Artificial Intelligence and Applications*, 4(5):49–61.

Zhengqi Pei, Zhewei Sun, and Yang Xu. 2019. Slang detection and identification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 881–889, Hong Kong, China. Association for Computational Linguistics.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. `http://is.muni.cz/publication/884893/en`.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Eleanor Rosch. 1975. Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, 104:192–233.

Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. Morfessor 2.0: Toolkit for statistical morphological segmentation. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics.

Karl Sornig. 1981. *Lexical Innovation: A Study of Slang, Colloquialisms, and Casual Speech*. John Benjamins B.V., Amsterdam.

Dan Sperber and Deirdre Wilson. 1986. *Relevance: Communication and Cognition*. Harvard University Press.

Zhewei Sun, Qian Hu, Rahul Gupta, Richard Zemel, and Yang Xu. 2024. Toward informal language processing: Knowledge of slang in large language models. In *Proceedings of the 2024 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 1683–1701, Mexico City, Mexico. Association for Computational Linguistics.

Zhewei Sun, Richard Zemel, and Yang Xu. 2019. Slang generation as categorization. In *Proceedings of the 41st Annual Conference of the Cognitive Science Society*, pages 2898–2904. Cognitive Science Society.

Zhewei Sun, Richard Zemel, and Yang Xu. 2021. A Computational Framework for Slang Generation. *Transactions of the Association for Computational Linguistics*, 9:462–478.

Zhewei Sun, Richard Zemel, and Yang Xu. 2022. Semantically informed slang interpretation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5213–5231, Seattle, United States. Association for Computational Linguistics.

Beatrice. Warren. 1992. *Sense Developments: A Contrastive Study of the Development of Slang Senses and Novel Standard Senses in English*. Acta Universitatis Stockholmiensis: Stockholm studies in English. Almqvist & Wiksell International.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Kilian Q. Weinberger and Lawrence K. Saul. 2009. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244.

Haryo Akbarianto Wibowo, Made Nindyatama Nityasya, Afra Feyza Akyürek, Suci Fitriany, Alham Fikri Aji, Radityo Eko Prasojo, and Derry Tanti Wijaya. 2021. IndoCollex: A testbed for morphological transformation of Indonesian colloquial words. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3170–3183, Online. Association for Computational Linguistics.

Ifeoluwa Wuraola, Nina Dethlefs, and Daniel Marciniak. 2024. Understanding slang with LLMs: Modelling cross-cultural nuances through paraphrasing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15525–15531, Miami, Florida, USA. Association for Computational Linguistics.

Tatu Ylonen. 2022. Wiktextract: Wiktionary as machine-readable structured data. In *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022)*, pages 1317–1325, Marseille, France.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

A Techniques for improving generative diversity

We evaluated the effectiveness of Chen et al. (2024) through an ablation test under default sampling parameters: temperature $= 1.0$ and top-$p = 1.0$. Comparing generation diversity in terms of the number of unique (w, sense) pairs across 1,000 generations under the U-F setting, we observed minimal improvement: 353 unique entries without the method versus 364 with it. This suggests that the technique does not substantially improve lexical and semantic diversity across batches.

B Computational setup for model distillation

In our configuration, we set the LoRA rank to 64, the LoRA scaling factor (alpha) to 128, and applied it across all linear layers with no dropout. The fine-tuning process follows a supervised fine-tuning (SFT) setup with a batch size of 32 and a maximum gradient norm of 1.0. A cosine learning rate scheduler was employed with an initial learning rate of $1\text{e}{-5}$ and no warm-up phase.

The number of training epochs was set to 5. Based on empirical observation, our models typically converges within 5 epochs. We found that extending training beyond 5 epochs yielded minimal returns in performance. Two Nvidia A6000 GPUs were used and each fine-tuning task took 16 mins.

C Experiment setup for downstream tasks

For each task, we sample 500 evaluation examples from each source. The OSD examples are sampled from the OSD test set (described in Section 4.3) and the OpenSub-Slang examples randomly sampled from the dataset.

Task 1 - Generation For every slang entry, we mask the target slang word in its usage example. The original word is treated as the correct answer, and three incorrect options are sampled randomly from the remaining entries in the test set. These four options are then randomly assigned to labels A, B, C, and D, with the correct label recorded. The definition and masked usage are combined with the prompt template (shown in Section E.1) to generate a multiple-choice question.

Task 2 - Interpretation Given a slang word and its usage context, we generate a multiple-choice question asking for its correct definition. The ground truth definition serves as the correct answer, while three incorrect definitions are randomly sampled from other entries in the test set. The options are randomly ordered and labeled A–D. The question prompt follows the format shown in Section E.2.

Task 3 - Free-form interpretation For each slang word and its usage example, we generate a free-form question asking the model to write an appropriate definition. The input prompt structure is specified in Section E.3, and evaluation is conducted by measuring semantic similarity (using SBERT) between the generated and ground-truth definitions.

D Prompt used to generate slang usages

D.1 U-F Prompt

```
You are a creative slang dictionary generator
    and here is the definition of slang:
A slang is a vocabulary (words, phrases, and
    linguistic usages) of an informal register,
    common in everyday conversation but avoided
    in formal writing and speech.
It also often refers to the language exclusively
     used by the members of particular in-groups
     in order to establish group identity,
    exclude outsiders, or both.
Generate novel slang usages in English.

The json structure must be:
{
  "word": [],       // An array of the slang
      terms
  "definition": [], // An array of corresponding
       definitions
  "usage_context": [] // An array of arrays,
      where each array has usage examples for
      that slang
}

- Keep the arrays aligned so the i-th element in
     "word", "definition", and "usage_context"
    all refer to the same slang and same length.
Remember:
- 'word' is the slang term.
- 'definition' is a short explanation.
- 'usage_context' should include 1-2 example
    sentences containing the slang term.

Now, generate {number_of_slang} entries.
The current dictionary already contains: [{
    existing_words}], do not repeat any of these
    .
```

D.2 U-R Prompt

```
You are a creative slang dictionary generator
    and here is the definition of slang:
A slang is a vocabulary (words, phrases, and
    linguistic usages) of an informal register,
    common in everyday conversation but avoided
    in formal writing and speech.
It also often refers to the language exclusively
     used by the members of particular in-groups
     in order to establish group identity,
    exclude outsiders, or both.
Generate novel slang usages in English.
'Generate' means taking existing English words
    and assigning them novel meanings to create
    novel slang, do not make up words.

The json structure must be:
{
  "word": [],       // An array of the slang
      terms
  "definition": [], // An array of corresponding
       definitions
  "usage_context": [] // An array of arrays,
      where each array has usage examples for
      that slang
}

- Keep the arrays aligned so the i-th element in
     "word", "definition", and "usage_context"
    all refer to the same slang and same length.
Remember:
- 'word' is the slang term.
- 'definition' is a short explanation.
- 'usage_context' should include 1-2 example
    sentences containing the slang term.

Now, generate {number_of_slang} entries.
The current dictionary already contains: [{
    existing_words}], do not repeat any of these
    .
```

D.3 U-C Prompt

```
You are a creative slang dictionary generator
    and here is the definition of slang:
A slang is a vocabulary (words, phrases, and
    linguistic usages) of an informal register,
    common in everyday conversation but avoided
    in formal writing and speech.
It also often refers to the language exclusively
     used by the members of particular in-groups
     in order to establish group identity,
    exclude outsiders, or both.
Generate novel usages in English.
'Generate' means creating novel words that do
    not exist in the conventional English
    lexicon.

The json structure must be:
{
  "word": [],       // An array of the slang
      terms
  "definition": [], // An array of corresponding
       definitions
  "usage_context": [] // An array of arrays,
      where each array has usage examples for
      that slang
}

- Keep the arrays aligned so the i-th element in
     "word", "definition", and "usage_context"
    all refer to the same slang and same length.
Remember:
- 'word' is the slang term.
- 'definition' is a short explanation.
- 'usage_context' should include 1-2 example
    sentences containing the slang term.

Now, generate {number_of_slang} entries.
The current dictionary already contains: [{
    existing_words}], do not repeat any of these
    .
```

D.4 C-F Prompt

```
You are a creative slang dictionary generator
    and here is the definition of slang:
A slang is a vocabulary (words, phrases, and
    linguistic usages) of an informal register,
    common in everyday conversation but avoided
    in formal writing and speech.
It also often refers to the language exclusively
     used by the members of particular in-groups
     in order to establish group identity,
    exclude outsiders, or both.
Generate novel slang usages in English to
    express the definition: {definition}.

The json structure must be:
{
  "word": [],       // An array of the slang
      terms
  "definition": [], // An array of corresponding
       definitions
  "usage_context": [] // An array of arrays,
      where each array has usage examples for
      that slang
}

- Keep the arrays aligned so the i-th element in
     "word", "definition", and "usage_context"
    all refer to the same slang.
Remember:
- 'word' is the slang term.
- 'definition' is the definition that is given.
- 'usage_context' should include at least 1-2
    example sentences containing the slang term.

Now, generate {number_of_slang} entries.
The current dictionary already contains: [{
    existing_words}], do not repeat any of these
    .
```

D.5 C-R Prompt

```
You are a creative slang dictionary generator
    and here is the definition of slang:
A slang is a vocabulary (words, phrases, and
    linguistic usages) of an informal register,
    common in everyday conversation but avoided
    in formal writing and speech.
It also often refers to the language exclusively
     used by the members of particular in-groups
     in order to establish group identity,
    exclude outsiders, or both.
Generate novel slang usages in English to
    express the definition: {definition}.
'Generate' means taking existing English words
    and assigning them the meaning to create
    novel slang, do not make up words.

The json structure must be:
{
  "word": [],       // An array of the slang
      terms
  "definition": [], // An array of corresponding
       definitions
  "usage_context": [] // An array of arrays,
      where each array has usage examples for
      that slang
}

- Keep the arrays aligned so the i-th element in
     "word", "definition", and "usage_context"
    all refer to the same slang and same length.
Remember:
- 'word' is the slang term.
- 'definition' is a short explanation.
- 'usage_context' should include 1-2 example
    sentences containing the slang term.

Now, generate {number_of_slang} entries.
The current dictionary already contains: [{
    existing_words}], do not repeat any of these
    .
```

D.6 C-C Prompt

```
You are a creative slang dictionary generator
    and here is the definition of slang:
A slang is a vocabulary (words, phrases, and
    linguistic usages) of an informal register,
    common in everyday conversation but avoided
    in formal writing and speech.
It also often refers to the language exclusively
     used by the members of particular in-groups
     in order to establish group identity,
    exclude outsiders, or both.
Generate novel slang usages in English to
    express the definition: {definition}.
'Generate' means creating novel words that do
    not exist in the conventional English
    lexicon.

The json structure must be:
{
  "word": [],       // An array of the slang
      terms
  "definition": [], // An array of corresponding
       definitions
  "usage_context": [] // An array of arrays,
      where each array has usage examples for
      that slang
}

- Keep the arrays aligned so the i-th element in
     "word", "definition", and "usage_context"
    all refer to the same slang and same length.
Remember:
- 'word' is the slang term.
- 'definition' is a short explanation.
- 'usage_context' should include 1-2 example
    sentences containing the slang term.

Now, generate {number_of_slang} entries.
The current dictionary already contains: [{
    existing_words}], do not repeat any of these
    .
```

E Task prompts

E.1 Task 1 – Generation

```
You are given a slang usage where the slang word
     has been masked with a blank (___), and a
     definition of that slang word.

Your task is to choose the correct slang word
     from the four options provided.

Usage:
{masked_usage}

Definition:
{definition}

Options:
A. {A}
B. {B}
C. {C}
D. {D}

Respond with a JSON object in the following
     format:
{
  "answer": "your answer in a single letter
      chosen from the options"
}

Only output the JSON object. Do not include any
     explanation.
```

E.2 Task 2 – Interpretation

```
You are given a slang word and a sentence
     showing how it's used in context.

Your task is to choose the correct definition of
     the slang word from the four options below.

Word:
{word}

Usage:
{usage}

Options:
A. {A}
B. {B}
C. {C}
D. {D}

Respond with a JSON object in the following
     format:
{
  "answer": "your answer in a single letter
      chosen from the options"
}

Only output the JSON object. Do not include any
     explanation.
```

E.3 Task 3 – Free-form interpretation

```
You are given a slang word and a sentence
    showing how it is used in context.

Your task is to write a concise definition of
    the slang word as it is used in this context
    .

Word:
{word}

Usage:
{usage}

Respond with a JSON object in the following
    format:
{
  "answer": "your concise definition here"
}

Only output the JSON object. Do not include any
    explanation.
```

E.4 Fine-tune corpus structure

```
Slang word: {word}\n
Defination: {definition}\n
Usage: {usage_context}\n
```

F Slang generation psudocode

F.1 Uncontrolled generation

Input:
Target number of slang entries N;
Existing slang entry set $\mathcal{E}$, where each entry is a unique tuple (w, senses);
Generation mode $m \in \{\texttt{Freeform}, \texttt{reuse}, \texttt{coinage}\}$;
Output:
An expanded slang set $\mathcal{E}$ with $|\mathcal{E}| = N$ unique entries

while $|\mathcal{E}| < N$ **do**
    Construct a prompt based on mode m and query the language model to generate candidate entries $\mathcal{C}$;
    foreach $(w, \textit{sense}, \textit{cxt}) \in \mathcal{C}$ **do**
        Classify w as either `coinage` or `reuse` using Wiktionary;
        if $m \neq \texttt{Freeform}$ ***and the classification does not match*** m **then**
            continue
        end
        if $(w, \textit{sense}) \notin \mathcal{E}$ **then**
            Add $(w, \text{sense}, \text{cxt})$ to $\mathcal{E}$;
        end
    end
end
return $\mathcal{E}$

Algorithm 1: Algorithm for uncontrolled slang generation

F.2 Controlled generation

Input:
Existing slang dictionary $D = \{(w, \text{sense}, \text{cxt})\}$ grouped by word sense;
Generation mode $m \in \{\texttt{Freeform}, \texttt{reuse}, \texttt{coinage}\}$;
Initial slang set $\mathcal{E}$ containing previously generated entries.
Output: A language model-generated dictionary D' with $|D'| = |D|$
Initialize $D' \leftarrow \emptyset$;
foreach *group* $d \subseteq D$ *corresponding to a unique word sense* **do**
  Initialize local slang set $\mathcal{E}_{\text{group}} \leftarrow \emptyset$;
  while $|\mathcal{E}_{group}| < |d|$ **do**
    Construct a prompt using word sense metadata of d and generation mode m;
    Query the language model to generate candidate entries $\mathcal{C}$;
    foreach $(w, \textit{sense}, \textit{cxt}) \in \mathcal{C}$ **do**
      Classify w as either `coinage` or `reuse` using Wiktionary;
      if $m \neq \texttt{Freeform}$ ***and*** *classification* $\neq m$ **then**
        continue ; `// Reject mismatched mode`
      end
      if $(w, \textit{sense}) \notin \mathcal{E}_{group} \cup d$ **then**
        Add $(w, \text{sense}, \text{cxt})$ to $\mathcal{E}_{\text{group}}$;
      end
    end
  end
  Append $\mathcal{E}_{\text{group}}$ to D';
end
return D'

Algorithm 2: Algorithm for controlled slang generation

G Coinage category classification psudocode

Input:
A dataset of slang words $\mathcal{D}$, where each word w is a candidate coinage;
Wiktionary index $\mathcal{W}$ mapping known words to definitions;
A trained Morfessor segmentation model $\mathcal{M}$;
Output:
A labeled dataframe $\mathcal{T}$ where each word is assigned a category label in `{Compound, Blend, Other}`
Initialize empty record list $\mathcal{T}$;
foreach *source group* $(s, W_s) \in \mathcal{D}$ **do**
  foreach *word* $w \in W_s$ **do**
    Segment w into subword units $S = \mathcal{M}.\texttt{segment}(w)$;
    if $|S| \geq 2$ **then**
      if *all* $s_i \in S$ *are exact matches in* $\mathcal{W}$ **then**
        Label w as `Compound`
      else if $s_1 \in S$ *is a preffix of some* $w' \in \mathcal{W}$ *and* $s_{-1} \in S$ *is a suffix of some* $w' \in \mathcal{W}$ **then**
        Label w as `Blend`
    else
      Label w as `Other`
    end
    Append $(s, w, |S|, \text{label})$ to $\mathcal{T}$
  end
end
return $\mathcal{T}$

Algorithm 3: Algorithm for classifying coinage types using Morfessor and Wiktionary

H Sample data

Table 8 shows examples from OSD and machine-generated slang usages.

I Topic analysis Full table

Table 9 shows the full results for the topic modeling experiment described in Section 4.1.

Source	Word	Definition	Usage Context
OSD	bruddah	alternate spelling of brother.	Safe, my bruddah.
OSD	cat off	Doing something out of the ordinary or stupid.	You cattin' off coming at me like that. Jerry went up to the girl to ask for a dance and she catted him off.
OSD	crop dust	to flatulate while walking through an area or by group of people.	Whoa! Smells like somebody has been crop dusting. He came in and crop dusted us.
OSD	cuckoo	crazy.	He's cuckoo.
OSD	cunt-fuck	to have vaginal sex.	My girlfriend and I got so wasted last night she asked me to cunt-fuck her.
C-C	zucchini zip	humorous word for a penis	He charmed everyone with tales of his zucchini zip. Expect his zucchini zip stories to get a chuckle from the crowd.
C-C	BeatBox808	A device that produces the signature sounds of the Roland 808.	The BeatBox808 was laying down a perfect bassline for the session. His set was on fire once he incorporated the BeatBox808 into the rhythm.
C-C	AUX	Acronym for "as you understand".	Can you give me a quick recap AUX? AUX, we're going for dinner at the usual spot.
C-C	AFUNU	Acronym for "as far as you know".	We're still meeting up later, AFUNU. AFUNU, they haven't decided on a location yet.
C-C	baebs	An endearing abbreviated form of "babe".	Hey baebs, want to grab some dinner tonight? How was your day, baebs?
C-F	scoot	A casual and informal way to indicate you are leaving.	Finished my work, I'm gonna scoot! It's getting late, time for me to scoot.
C-F	sloshed	Extremely drunk, to the point of losing control.	He was so sloshed he couldn't even walk straight. She was sloshed after the party and had to take a cab home.
C-F	you bunch	A casual way of referring to everyone present or being spoken to.	You bunch better be ready for the game tonight! Where's the energy, you bunch? Let's get hyped up!
C-F	nugget	A small, cute term for a breast.	Her shirt was tight, revealing the outline of a nugget. A gentle pat on her nugget was met with a playful smack.
C-F	SAGZ	An acronym for Sex, Age, Gender, Zodiacs.	Instead of the usual small talk, she popped a SAGZ. When someone asks me for my SAGZ, it makes the chat more engaging. Try it next time!
C-R	day one	A best friend who has been there from the beginning.	He's my day one, always there since the beginning. I can trust her with anything; she's been my day one.
C-R	hubcap	A term for your significant other who keeps your world running smoothly.	Whenever things get hectic, I know my hubcap is there to keep everything together. She's not just my partner, she's my hubcap making the everyday run smoothly.
C-R	spin the bottle	To perform fellatio.	I think she's going to spin the bottle with him later. Let's see who's brave enough to spin the bottle tonight.
C-R	dig deep	To thoroughly review or research prior studies.	The student had to dig deep into past studies to find the missing link in his research. To fully understand the context, I needed to dig deep into prior academic journals.
C-R	memory jogger	An action to stir recollections or awareness.	Her old friend's visit acted like a memory jogger, bringing back countless memories. After procrastinating all day, the looming deadline was a real memory jogger.
U-C	blizzleplunk	A sudden change in direction during a walk or drive.	When he turned that corner, it was a blizzleplunk all the way! Our walk turned into a blizzleplunk after he got us lost.
U-C	splogboop	An unexpected delightful surprise	I found a $20 bill on the street today, total splogboop!
U-C	blizzlefrost	A cold, frosty chill of excitement	The first snowflake of the season gave me such a blizzlefrost.
U-C	trungleflap	To haphazardly bounce or tumble	Watch out, don't trungleflap over the rug!
U-C	zorkmingle	A quirky social gathering	We went to a zorkmingle at Jane's place last night.
U-F	fluffle	A cozy group of cute or fluffy things gathered together.	The fluffle of kittens was too cute to handle. Nothing beats a fluffle of bunnies in the morning to lift your spirits.
U-F	doomscroll	The act of endlessly scrolling through bad news.	I lost two hours to doomscroll on Twitter last night. To break the cycle, I've installed an app to curb my doomscroll habit.
U-F	jugglework	The complex act of balancing multiple tasks at work.	Working in marketing means constant jugglework, especially during campaigns. She manages her jugglework skillfully, balancing three roles seamlessly.
U-F	techtime	Quality screen time for relaxation or productivity.	We've scheduled techtime to catch up on some documentaries this weekend. Let's have a techtime session to binge some classics.
U-F	cringeflash	The rush of secondhand embarrassment from awkward memories.	I experienced a cringeflash when I remembered my failed public speech in high school. Every time that song plays, I get a cringeflash of my awkward dance moves.
U-R	backwash	The residual effects of an event or situation.	After the festival, there was a backwash of positive energy and camaraderie. The media backwash from the announcement was overwhelming.
U-R	cinderblock	Solid and unmovable, like firm determination.	Her determination was like a cinderblock, unyielding and strong. We need a cinderblock of confidence to get through this challenge.
U-R	switchblade	A quick, witty comeback or response.	After a bit of banter, Mike unleashed his switchblade that left everyone speechless. Her quick switchblade during the discussion won her a lot of applause.
U-R	lanternfish	Someone who is a night owl.	Kyle, the lanternfish of the group, is always busy when the rest of us are sleeping. The neighborhood knows Sam as a lanternfish because his lights are always on at midnight.
U-R	bathrobe	The state of feeling relaxed and at ease.	After a long week, the weekend felt like slipping into a bathrobe, comfortable and warm. Whenever I'm stressed, talking to Jenny is like putting on a bathrobe.

Table 8: Randomly sampled examples from both GPT-generated slang usages and OSD.

Topic	OSD Topic Words	GPT-4o Topic Words	LLaMA-8B Topic Words
1	adjective, get, person, term, shit, acronym, think, guy, bad, dude	quick, unexpected, reaction, laughter, cause, change, catch, news, mind, situation	new, party, find, flumplen, add, decorate, group, night, idea, search
2	go, time, fuck, want, ass, way, good, work, think, say	idea, give, make, plan, subtle, creativity, project, look, new, thought	new, need, ing, kid, catch, friend, jargle, playful, stop, love
3	female, give, male, need, boy, ride, movie, face, girlfriend, little	excitement, sudden, unexpected, moment, energy, meeting, cause, intense, feel, room	feel, get, energy, feeling, dance, concert, start, crowd, party, excitement
4	look, night, uncountable, person, go, sex, guy, cool, number, play	day, gentle, feel, party, light, movement, room, energy, lively, add	try, way, friend, good, hour, get, snurfle, work, flumplen, end
5	person, man, find, save_deletion_legitimate_citation, definition_questionable, pende_deletion, let, woman, get, come	surprise, leave, excitement, dance, sudden, movement, action, event, energy, unexpected	flumplen, look, try, situation, team, snurfle, person, new, take, room

Table 9: Top-10 LDA topic words from both OSD, GPT-4o, and Llama-3-8B sense definitions.