Lexical Conditioning of Model's Distribution through Uncertainty-gated Soft-Mixing of Probabilities

Michele Papucci[1,2], **Giulia Venturi**[2], **Felice Dell'Orletta**[2]
[1]University of Pisa
[2]ItaliaNLP @ Institute for Computational Linguistics "A. Zampolli" (CNR-ILC), Pisa
michele.papucci@phd.unipi.it, {giulia.venturi, felice.dellorletta}@ilc.cnr.it

Abstract

We present Uncertainty-Gated Lexical Decoding (UGLD), a decoding-time framework for fine-grained lexical control in Large Language Models (LLMs) that explicitly addresses the trade-off between controllability and fluency. UGLD adaptively scales intervention through an entropy-based gating mechanism derived from the model's predictive distribution, activating control when uncertainty is high and limiting interference when predictions are confident. The method supports both promotion toward and against predefined vocabularies. We evaluate UGLD in Italian on two open-weight LLMs (ANITA 8B and Qwen 3 4B) across paraphrasing and free-text generation settings, considering Simple Vocabulary Conditioning and Jargon Reduction scenarios. Automatic evaluation shows consistent improvements in lexical coverage over standard decoding strategies, while human evaluation confirms that fluency is preserved under controlled intervention.

Keywords: Controlled Text Generation, Lexically Constrained Decoding, Entropy-Gated Decoding

1. Introduction

Controlling the behavior of Large Language Models (LLMs) for Text Generation is becoming increasingly important as these models are used in a growing number of real-world scenarios that require adherence to multiple constraints. To address this need, recent techniques for Controlled Text Generation (CTG) include fine-tuning approaches (Nguyen et al., 2024), prompt-based methods that express the constraint as a natural-language instruction (Zhou et al., 2023), and weighted decoding strategies that intervene directly in the model's output distribution to adjust token probabilities during the decoding stage (Pascual et al., 2021; Yang and Klein, 2021). While the latter approaches are particularly suitable, especially in low-resource settings, they face the recurrent challenge of maintaining fluency of the generated texts. In fact, prior work has shown that as the control strength increases beyond a certain threshold, fluency can rapidly decrease (Zhong et al., 2023).

Building on these premises, we introduce *Uncertainty-Gated Lexical Decoding*, a decoding-time approach for fine-grained lexical control that aims to preserve the fluency of the generated text. The core idea is to modulate the strength of decoding interventions according to the model's uncertainty in next-token prediction, rather than applying lexical constraints uniformly across the generation process. To this end, we devise an *uncertainty-gated* mechanism derived from the entropy of the model's predictive distribution, which activates control primarily when the model is uncertain and limits interference when it is confident in its predictions. The gate supports two complementary forms of lexical control: conditioning generation *towards* a predefined set of lexical items through explicit lexical priors, and conditioning it *against* undesired vocabulary through logit-level penalties that suppress specific words.

We evaluate the proposed approach in two CTG settings: **paraphrasing**, where the model rewrites an input sentence under lexical constraints, and **free-text generation**, where constraints are applied during unconstrained continuation. Experiments are conducted on Italian and focus on two complementary forms of lexical conditioning that are relevant in real-world applications: **Simple Vocabulary Conditioning**, which involves guiding generation towards a predefined lexicon of simple and high-accessibility words, and **Jargon Reduction**, which aims to guide generation away from domain-specific terminology by suppressing technical lexical items. We consider these two conditioning scenarios because they represent complementary directions of lexical control. Together, they can be viewed as building blocks that could be integrated into a broader Controlled Text Simplification framework, which we leave as future work.

Contributions: *i)* we propose a novel decoding-time framework that adaptively modulates lexical control through an entropy-based gating mechanism; *ii)* we make use of the lexical control mechanism to condition two LLMs towards and against a predefined vocabulary, without requiring additional training or external discriminators, making it particularly suitable for low-resource settings and

Proceedings of the Joint Workshop on Readability and Text Simplification (READIxTSAR) @ LREC 2026, pages 89–100
11 May 2026.

languages[1]; *iii)* we provide a twofold evaluation consisting of an automatic coverage metric to quantify lexical shifts across different levels of intervention strength and human judgments to assess fluency preservation.

2. Related Works

Decoding techniques have been used as a way to control LLMs' generation, and are widely considered a family of Controlled Text Generation (CTG) techniques (Zhang et al., 2023). Relevant previous works fall broadly into hard-constraint or soft-constraint decoding methods.

Hard constraint decoding methods, such as Grid Beam Search and Dynamic Beam Allocation (Hokamp and Liu, 2017; Post and Vilar, 2018), enforce the presence of specific words or phrases through structural modifications of the beam search. While effective at guaranteeing constraint satisfaction, they do not provide adaptive control, leading to less fluent text.

Soft-constraint approaches, including GeDI, PPLM, FUDGE, and DExperts (Krause et al., 2021; Pascual et al., 2021; Yang and Klein, 2021; Liu et al., 2021), modify token probability to induce high-level attributes (e.g., sentiment, toxicity, topic). These methods, however, rely on a fixed strength intervention and/or on a 'learned' discriminator to manipulate the model's logits. Such discriminators are often implemented as trained classifiers, which require additional annotated data and training resources that may not be available in low-resource settings, or as auxiliary LLMs, which substantially increase computational cost at inference time. Our method instead uses *explicit lexical priors* and *logit penalties* with an *adaptive intervention strength modulated by the model's own uncertainty*, and does not require training or querying external models. More generally, incorporating uncertainty into decoding decisions has been explored in prior work through entropy-based measures, which provide a model-internal signal of confidence during language generation, although these approaches are not typically formulated within a CTG setting. Locally Typical Sampling (Meister et al., 2023) restricts the sampling to tokens whose information content matches the model's conditional entropy to mimic the human information pacing. Similarly, entropy-based methods such as EGED (Das et al., 2025) adjust decoding behavior under uncertainty, but do not attempt lexical or attribute steering. ECO decoding (Shin et al., 2025) is closest to our method: it adapts attribute-control strength using entropy, but still assumes a classifier-based attribute model rather than explicit vocabularies.

3. Methodology

We present a decoding technique for Controlled Text Generation called **Uncertainty-Gated Lexical Decoding** (UGLD). In particular, we present two variations of UGLD: one that conditions the model's distribution towards producing a pre-set vocabulary of tokens (UGLD-t), the other conditions the model's distribution against the chosen vocabulary (UGLD-a). To do that, while preserving the model's fluency, we scale the intervention strength by the model's uncertainty in the next-token prediction, leading to strong conditioning only when the model is uncertain. To model uncertainty, we employ Shannon's Entropy $H(\mathbf{p})$:

$$H(\mathbf{p}) = -\sum_i p_i \log p_i$$

where $\mathbf{p}$ is a probability vector of the size of the model's vocabulary $\mathcal{V}$, obtained by soft-maxing the last layer model logits $\mathbf{z}$, i.e., $\mathbf{p} = \text{SoftMax}(\mathbf{z})$ before applying any probability manipulations.
By using the entropy $H(\mathbf{p})$ we define a smooth gating mechanism $\phi(\mathbf{p}) \in [0, 1]$, that determines how strongly to condition the distribution:

$$\phi(\mathbf{p}) = \sigma(\frac{H(\mathbf{p}) - \tau}{s}) \quad (1)$$

Here, τ acts as an entropy threshold when the intervention is activated. For $H(\mathbf{p}) \ll \tau$, the gate is closed $\phi \approx 0$; for $H(\mathbf{p}) \gg \tau$ the gate is open $\phi \approx 1$. Since $H(\mathbf{p}) \in [0, \log|\mathcal{V}|]$, τ is typically chosen in $[0, \log|\mathcal{V}|]$, as outside this range we have degenerate behaviors[2]. Finally, σ is the sigmoid function that normalizes the gate $\phi(\mathbf{p}) \in [0, 1]$ and s is a strictly positive smoothing factor that controls *how fast* the gate opens: a smaller s value makes the sigmoid reach 1 with lower values of $H(\mathbf{p})$.

Conditioning Towards (UGLD-t) Given a subset of tokens that we want the LLM to generate with **higher** relative probability than the rest of the vocabulary, hereafter referred to as *green tokens* ($\mathcal{V}_{green} \subset \mathcal{V}$), we employ a soft mixture of probability distributions to shift the model's output distribution toward the green tokens when the uncertainty is high:

$$\mathbf{p}' = (1 - \alpha)\mathbf{p} + \alpha\mathbf{q} \quad (2)$$

Here $\mathbf{p}$ and $\mathbf{q}$ are the two probability distributions we are mixing. In particular, $\mathbf{p}$ is the probability

[1] UGLD is available on GitHub: https://github.com/michelepapucci/ugld and can be installed with PyPI: https://pypi.org/project/ugld/

[2] When $\tau \gg \log|\mathcal{V}|$ we approximate an always closed gate, and with $\tau \ll 0$ we approximate an always open gate

vector over the vocabulary produced at the current decoding step, while $\mathbf{q}$ is our conditioning prior distribution that allocates all of its probability mass to the *green* tokens. Formally, $\mathbf{q}$ is any valid probability distribution where $q_i = 0 \; \forall i \notin \mathcal{V}_{green}$ and $\sum_i q_i = 1$. Finally, α represents the intervention strength, calculated as the maximum intervention strength allowed $\alpha_{max} \in [0, 1]$ weighted by the entropy gate $\phi(\mathbf{p})$:

$$\alpha = \phi(\mathbf{p})\alpha_{max}$$

Because $\alpha \in [0, 1]$, this update forms a convex combination of the two distributions, ensuring that $\mathbf{p}'$ remains a valid probability distribution. Possible instantiations of the conditioning distributions $\mathbf{q}$ are described in Section 4.2.

Conditioning Against (UGLD-a) Given a subset of tokens that we want the LLM to generate with **lower** relative probability than the rest of the vocabulary, hereafter referred to as *red tokens* ($\mathcal{V}_{red} \subset \mathcal{V}$), we manipulate the model-produced logits to penalize the generation of those tokens when the uncertainty is high. To avoid normalization artifacts or negative probabilities, the negative conditioning is performed directly in **logits space**. Given the model's logits $\mathbf{z}$, we can subtract a vector of penalties $\lambda\mathbf{r}$:

$$\mathbf{z}' = \mathbf{z} - \lambda\mathbf{r}$$

where $\mathbf{r}$ is a vector of size $|\mathcal{V}|$ that contains weights for each token inside the *red* vocabulary. Specifically, $r_i > 0 \; \forall i \in \mathcal{V}_{red}$ and $r_i = 0$ otherwise. λ is instead the penalty strength that gets scaled for each token by $\mathbf{r}$, and is calculated as the maximum penalty strength allowed $\lambda_{max} \geq 0$ weighted by the entropy gate $\phi(\mathbf{p})$:

$$\lambda = \phi(\mathbf{p})\lambda_{max}$$

The newly obtained logits $\mathbf{z}'$ can be SoftMaxed to obtain the final probability distribution over the vocabulary from which the next-token prediction can be sampled. Possible ways to build the weight vector $\mathbf{r}$ are described in Section 4.2.

4. Experimental Settings

The effectiveness of the Uncertainty-Gated Lexical Decoding strategy was evaluated in two conditioning scenarios. The Conditioning Towards strategy was tested in a **Simple Vocabulary Conditioning** scenario, which evaluates the ability of the strategy to guide the model's generation towards a predefined set of simple and accessible words, while the Conditioning Against strategy was tested in a **Jargon Reduction** scenario, which evaluated the ability to guide generation away from domain-specific terminology by suppressing technical vocabulary. In both scenarios, the decoding strategy was tested in two controlled text generation settings: **paraphrasing**, where an original sentence is rewritten under lexical conditioning, and **free-text generation**, where lexical conditioning is applied while generating a short paragraph as a continuation of an input sentence[3].

All experiments were conducted using two open-weight LLMs: one fine-tuned for Italian and one multilingual. Specifically, we considered ANITA 8B (Polignano et al., 2024)[4], a model fine-tuned on Italian data and based on LLaMA 3 (Grattafiori et al., 2024), and Qwen 3 4B (Yang et al., 2025)[5], a multilingual LLM pre-trained on large-scale multilingual corpora, supporting Italian generation and used here in its instruction-tuned version.

4.1. Data and Lexical Resources

Simple Vocabulary Conditioning. We considered a set of 100 sentences randomly sampled from Wikipedia page dumps. We selected Wikipedia as our source data because it provides a heterogeneous collection of general-purpose texts. The two tested LLMs were conditioned toward generating *green tokens* from the *New Basic Italian Vocabulary* (NBIV) (De Mauro and Chiari, 2016), designed to include the vocabulary that ensures high readability for readers with heterogeneous education levels and originally including a list of 7,000 words highly familiar to native speakers of Italian. Specifically, we considered the subset of words belonging to the class of *Fundamental Words*, which includes very frequent and widely used words in Italian. Within this class, we included in the list of conditioning green tokens only words categorized as adjectives, nouns, verbs, and past and present participles. This choice is motivated by the intuition that content words, rather than grammatical ones, carry the core semantic load of a sentence and therefore represent the most effective targets for lexical conditioning aimed at increasing accessibility. To convert this list from a list of lemmas to a list of tokens, we first expanded each lemma to every possible form, obtaining a final list of $98\,430$ word forms, then converted it to token IDs using the tokenizer for the selected model, obtaining 4686 tokens for Anita and 4654 tokens for Qwen.

Jargon Reduction. Although the scenario is in principle domain-general, we adopt the medical do-

[3] This was achieved through prompting. For the paraphrasis, we asked the model to rewrite the sentence in its own words. while for free-text generation we asked the model to continue the sentence with a short paragraph.

[4] swap-uniba/LLaMAntino-3-ANITA-8B-Inst-DPO-ITA

[5] Qwen/Qwen3-4B-Instruct-2507

main as representative, as it has been widely considered in studies aimed at reducing the complexity of expert-specific jargon. To this end, we used the Italian version of the manuals available on MSD Manuals[6], one of the world's most widely used specialized websites offering publicly accessible medical information for both healthcare professionals (Professional version) and non-expert users (Consumer version), written to be clear and accessible to non-expert readers. We considered a total of 100 sentences. As a lexical resource, we consider the lexicon deriving from the Professional and Consumer versions of the MSD articles. Specifically, we employed a contrastive TF-IDF variant. We computed Term Frequency (TF) on the professionals' corpus by aggregating and length-normalizing term counts across all professional documents. We then computed Inverse Document Frequency (IDF) on the consumers' corpus, measuring how many consumer documents each term appears in (with smoothing). Candidate terms were extracted from professionals' texts and ranked by multiplying their professional TF by their consumer-based IDF. Intuitively, this procedure extracts words that are frequent in professionals' texts but rare across consumers' documents. In practice, it highlights terminology that is characteristic of the professional domain, such as technical, clinical, or specialized expressions, while downweighting words that are common in both groups. Finally, we expanded each word to all possible forms of its lemma, obtaining $111\,196$ forms, and extracted the token IDs using the tokenizer for the selected model, yielding $11\,820$ tokens for Anita and $11\,727$ for Qwen.

4.2. UGLD Hyperparameter Choice

To test UGLD in both scenarios, we had to choose a number of hyperparameters.

Simple Vocabulary Conditioning. For this scenario, we employed UGLD-t, to condition the LLMs towards generating more tokens that appear in the NBIV. For the choice of conditioning prior $\mathbf{q}$ (See Eq. 2), we experimented with three possible choices:

1. **Uniform** We build a uniform prior that spreads the probability mass equally across all green tokens in $\mathcal{V}_{green}$:

$$q_i = \begin{cases} \frac{1}{|\mathcal{V}_{green}|} & i \in \mathcal{V}_{green} \\ 0, & \text{otherwise} \end{cases}$$

2. **Top-K** We create a uniform prior that only spreads its probability mass across the top-K candidates in V_{green} in terms of probability of being produced at that time step:

$$q_i = \begin{cases} \frac{1}{|K|} & i \in \text{Top-}K(\mathcal{V}_{green}) \\ 0, & \text{otherwise} \end{cases}$$

3. **Re-normalizing** We re-normalize the model probability distribution at each time step over the set of green tokens V_{green}:

$$q_i = \begin{cases} \frac{p_i}{\sum_{j \in \mathcal{V}_{green}} p_j} & i \in \mathcal{V}_{green} \\ 0, & \text{otherwise} \end{cases}$$

We then set the threshold τ based on the distribution of entropy values $H(\mathbf{p})$ in the dataset (See Figure 1a) so that the entropy is higher than τ in around a third of the decoding steps. We chose 0.1 for Qwen, and 0.5 for Anita. Then, for α_{max} and s we tested each prior in three different scenarios:

- **Soft Conditioning**. We kept the intervention soft, with a max intervention strength $\alpha_{max} = 0.25$ and a relatively fast smoothing $s = 0.3$;
- **Medium Conditioning**. Here we increased the max intervention strength $\alpha_{max} = 0.5$ and kept the smoothing $s = 0.3$;
- **Strong Conditioning**. Finally, we pushed the intervention strength to $\alpha_{max} = 0.8$ and made the decoding transition very rapidly towards the maximum strength of intervention with a very fast smoothing $s = 1^{-12}$;

Jargon Reduction. For the task, we employed UGLD-a to condition the model to generate fewer technical words. To do that, we experimented with two different ways of constructing the weight vector $\mathbf{r}$:

1. **Fixed Weights**: A fixed penalty weight of 1 is applied to each token in V_{red}:

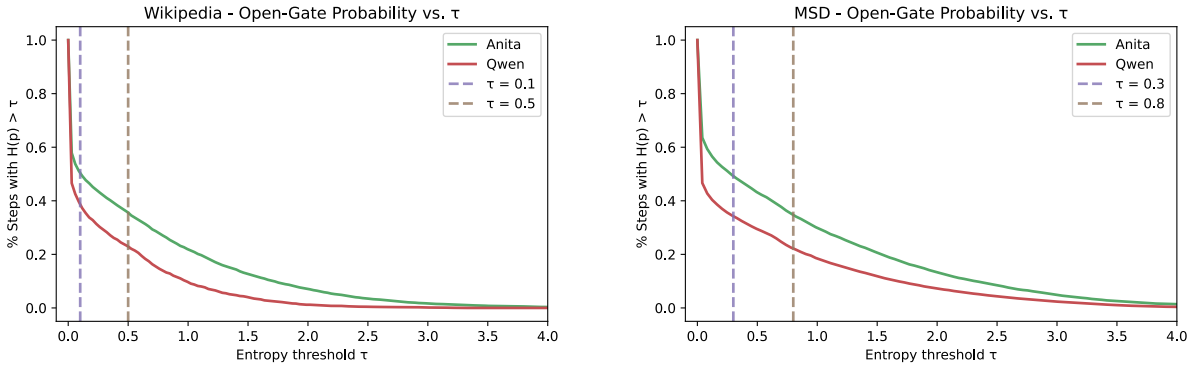
$$r_i = \begin{cases} 1 & \text{if } i \in R \\ 0 & \text{otherwise} \end{cases}$$

2. **Dynamic Weights**: A dynamic penalty is calculated at each time step, where a larger penalty is allocated towards red tokens that the model is currently more likely to generate. For each $i \in |\mathcal{V}_{red}|$ we compute a relative score via min–max normalization over the probabilities of red tokens p^{red} and map it to $[1, 2]$:

$$r_i = \begin{cases} 1 + \frac{p_i - \min(p^{red})}{\max(p^{red}) - \min(p^{red})} & \text{if } i \in \mathcal{V}_{red} \\ 0 & \text{otherwise} \end{cases}$$

Similar to the Simple Vocabulary Conditioning scenario, we fixed τ to 0.3 for Qwen and 0.8 for

[6] https://www.msdmanuals.com



(a) Wikipedia

(b) MSD

Figure 1: Percentage of decoding steps with entropy higher than possible τ threshold values. On the left, the entropy calculated on Wikipedia texts, on the right, the entropy calculated on MSD texts.

Anita based on the model's entropy distribution over the decoding steps of the dataset (see Figure 1b). Then, for λ_{max} and s we created again three scenarios:

- **Soft Conditioning**. We tried a soft penalty to the logits, with a $\lambda_{max} = 2$ and a relatively fast smoothing $s = 0.3$;
- **Medium Conditioning**. Here we increased the max penalty $\lambda_{max} = 4$ and kept the smoothing $s = 0.3$;
- **Strong Conditioning**. Finally, we pushed the penalty $\lambda_{max} = 6$ and made the decoding transition very rapidly towards the maximum penalty, with a very fast smoothing $s = 1^{-12}$;

4.3. Evaluation Methods

We adopted two types of evaluation methods, i.e., automatic metrics and human judgments, aimed at assessing complementary aspects of the UGLD approach. Automatic metrics are considered to quantify the extent to which the approach successfully guides the model's lexical choices toward or away from predefined vocabularies under different decoding settings. On the contrary, human judgments are aimed at assessing fluency of the generated text, given that decoding-time control may negatively affect this aspect of the generation process.

Automatic evaluation. For both controlled generation settings (paraphrasing and free-text generation), we computed the coverage of either green or red tokens in the generated outputs under different conditioned decoding strategies, considering both greedy decoding and nucleus sampling. Specifically, The coverage is calculated as the length-normalized percentage of generated tokens that are part of the selected lexicon, so for the the Simple Vocabulary Conditioning it is the proportion of generated green tokens belonging to the target accessible lexicon[7] while for Jargon Reduction, we measured the proportion of generated red tokens belonging to the technical vocabulary. For each setting, we compared the baseline decoding behavior (Greedy and Nucleus) with its uncertainty-gated counterpart (+ UGLD-t or + UGLD-a), ensuring that observed differences can be attributed to the proposed conditioning strategy rather than to decoding variability.

Human judgments. While automatic evaluation was conducted on the full set of experimental configurations, human judgments were collected on a selected subset. We focused on the Simple Vocabulary Conditioning scenario, as it involves paraphrasing and free-text generation of Wikipedia-style sentences, which are more easily assessable by human evaluators than texts from specialized medical domains. Human evaluation was further restricted to comparing the Nucleus baseline with Nucleus+UGLD-t under the Re-normalizing configuration. This choice was made as the Nucleus baseline is a very commonly used decoding strategy used in the wild. As for the other hyperparameters, we considered only the Soft and Strong conditioning settings, corresponding to the two extremes of conditioning strength, in both paraphrasing and free-text scenarios. This resulted in 50 pairs per model (Qwen and ANITA) per scenario, for a total of 400 pairs. For each model and scenario, the pairs were randomly distributed into 2 questionnaires of 25 pairs each, administered to 5 distinct annotators recruited via Prolific, all native speakers of Italian, for a total of 80 annotators[8]. For each pair,

[7] Table 6 (in the Appendix) shows an example in paraphrasing a sentence using different models and different UGLD-t configurations.

[8] Annotators were compensated £7.50/h. For the paraphrasis setting, we estimated a completion time of 20 min. while the observed average completion time was

Green Token Coverage on Wikipedia (Paraphrasis) ↑

Original Text		0.37								
Anita	**Greedy**	0.38								
	Nucleus	0.37								
		Uniform			**Top-K**			**Re-Normalization**		
		Soft	Medium	Strong	Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-t	0.38	0.38	0.38	0.38	0.38	0.40	0.38	0.38	**0.44**
	Nucleus + UGLD-t	0.37	0.37	0.36	0.40	0.40	0.40	0.39	0.39	0.40
Qwen	**Greedy**	0.38								
	Nucleus	0.37								
		Uniform			**Top-K**			**Re-Normalization**		
		Soft	Medium	Strong	Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-t	0.38	0.38	0.38	0.38	0.38	0.40	0.39	0.39	**0.42**
	Nucleus + UGLD-t	0.37	0.38	0.37	0.37	0.38	0.37	0.39	0.38	0.41

Table 1: In gray the best greedy configuration, in blue is the best nucleus configuration. In **bold** the best configuration overall, per-model.

annotators answered 2 binary (yes/no) questions assessing the fluency of each sentence in terms of grammatical correctness ("Is sentence 1 grammatically correct?" / "Is sentence 2 grammatically correct?"). The UGLD and Nucleus sampled sentences came from the same prompt, model, and scenario, with only the decoding strategy changed, and were randomly assigned to be either sentence 1 or sentence 2.

5. Results

Automatic evaluation. Results are reported in Tables 1, 2, 3, and 4. Tables 1 and 2 present results for the paraphrasing setting, reporting lexical coverage in the conditioning-toward scenario (measured as green-token coverage) and the conditioning-against scenario (measured as red-token coverage) for Wikipedia and MSD texts, respectively. Tables 3 and 4 report the corresponding results for free-text generation. For all settings, we consider greedy decoding and nucleus sampling as baselines. The latter coverage was obtained as the average coverage across 9 generation runs. We then applied UGLD-t to Wikipedia and UGLD-a to MSD, and after manipulating the probability distribution with the two techniques, we decoded both by greedy decoding (Greedy + UGLD) and nucleus decoding (Nucleus + UGLD)[9].

As a general outcome, we can observe that **free-text generation exhibits stronger lexical control effects than paraphrasing**. Specifically, free-text generation yields higher green token coverage in the conditioning-toward scenario (UGLD-t) and lower red token coverage in the conditioning-against scenario (UGLD-a). This pattern is consistent with the entropy-based design of UGLD: free-text generation typically involves higher uncertainty, which increases the activation of the gating function and results in stronger lexical control. In contrast, paraphrasing constrains the model through the source sentence, reducing uncertainty and therefore limiting the effect of the conditioning.

More specifically, we can observe a number of differences across conditioning scenarios. In the conditioning-toward scenario (Wikipedia), baseline green-token coverage in paraphrasing remains close to that of the original sentence for both ANITA and Qwen. Under UGLD-t, modest gains are observed with Uniform and Top-K priors, while Re-Normalization yields more substantial improvements, particularly when combined with greedy decoding. A similar trend emerges in free-text generation, where stronger hyperparameter settings further increase green-token coverage for both models. In contrast, in the second conditioning scenario (MSD), conditioning against the red vocabulary (UGLD-a) produces markedly larger shifts relative to baseline, for both models. Although Nucleus paraphrasing already reduces red-token coverage compared to the original text, UGLD-a further decreases it for both models, especially when combined with Dynamic Weights under Strong hyperparameter settings. In free-text generation, this effect becomes even more pronounced. Overall, these results indicate that **conditioning against a vocabulary produces larger coverage differences** than conditioning toward a vocabulary, across both generation settings. It seems to suggest that suppression tends to induce larger shifts because directly penalizing selected tokens forces a stronger redistribution of token probabilities, whereas promotion must compete with the model's original token ranking, resulting in more moderate deviations from

21 minutes and 24 sec. For the free-text setting, we estimated a completion time of 30 min. while the observed average completion time was 32 minutes and 39 sec.

[9]We chose a top-p = 0.9, which is a common choice for the nucleus decoding

Red Token Coverage on MSD (Paraphrasis) ↓

Original Text		0.44					
Anita	**Greedy**	0.41					
	Nucleus	0.39					
		Fixed Weights			**Dynamic Weights**		
		Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-a	0.38	0.39	0.39	0.39	0.40	0.41
	Nucleus + UGLD-a	0.35	0.31	0.24	0.32	0.25	**0.22**
Qwen	**Greedy**	0.44					
	Nucleus	0.43					
		Fixed Weights			**Dynamic Weights**		
		Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-a	0.43	0.43	0.43	0.43	0.43	0.43
	Nucleus + UGLD-a	0.41	0.39	0.34	0.40	0.35	**0.29**

Table 2: In gray the best greedy configuration, in blue is the best nucleus configuration. In **bold** the best configuration overall, per-model.

Green Token Coverage on Wikipedia (Free-Text generation) ↑

Anita	**Greedy**	0.39							
	Nucleus	0.39							
		Uniform			**Top-K**			**Re-Normalization**	
		Soft	Medium	Strong	Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-t	0.39	0.39	0.39	0.40	0.41	**0.43**	0.40	0.42	0.42
	Nucleus + UGLD-t	0.39	0.39	0.39	0.40	0.40	0.39	0.40	0.41	**0.43**
Qwen	**Greedy**	0.47							
	Nucleus	0.47							
		Uniform			**Top-K**			**Re-Normalization**	
		Soft	Medium	Strong	Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-t	0.47	0.47	0.47	0.48	0.48	**0.50**	0.48	**0.50**	**0.50**
	Nucleus + UGLD-t	0.46	0.47	0.46	0.48	0.47	0.47	0.47	0.49	**0.50**

Table 3: In gray the best greedy configuration, in blue is the best nucleus configuration per model. In **bold** the best configuration overall, per-model.

the baseline.

A last remark concerns the hyperparameters setting. As we can see, across models and conditioning scenarios, the **Re-Normalization and Dynamic Weights configurations tend to yield the strongest lexical control**. This suggests that more adaptive strategies, those that dynamically adjust the intervention at each decoding step, are more effective than static priors in shaping lexical choice. Moreover, the **Strong configuration produces the largest coverage shifts**, reflecting the impact of higher intervention strength.

Human evaluation. Table 5 reports inter-annotator agreement (Fleiss' κ), calculated separately for the first (S1) and second (S2) sentence (or text) in each evaluated pair, and fluency rates (Fluency %) for paraphrasing and free-text generation under baseline nucleus decoding and the uncertainty-gated counterpart (Nucleus+UGLD-t), for both ANITA and Qwen. All evaluations were conducted on outputs generated under the Soft and Strong conditioning strengths. Fluency percentages correspond to the proportion of 50 sentences/texts, per configuration, for which the majority of annotators answered "Yes" (baseline vs. UGLD).

To test whether UGLD affects fluency, we apply the exact McNemar test on paired binary judgments. This test is designed for matched outputs (baseline vs. UGLD for the same input) and evaluates whether the two systems differ in the proportion of positive judgments, focusing on discordant pairs. Across all settings, differences in fluency between UGLD and Nucleus are small, and all of them are non-significant ($p > 0.05$), indicating that UGLD does not systematically degrade grammatical correctness. Even in the Strong conditioning setting, where lexical coverage shifts are largest, fluency remains statistically comparable to baseline decoding. In some configurations, UGLD is numerically higher, though not significantly so.

As we can see, agreement scores range from low to moderate, consistent with prior work on subjective fluency judgments, even among native speakers, as annotators may vary in whether they focus on strictly grammatical aspects or also consider broader orthographic and stylistic factors. Impor-

Red Token Coverage on MSD (Free-Text generation) ↓							
Anita	**Greedy**	0.39					
	Nucleus	0.38					
		Fixed Weights			**Dynamic Weights**		
		Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-a	0.39	0.38	0.39	0.38	0.37	0.38
	Nucleus + UGLD-a	0.35	0.28	0.20	0.31	0.19	**0.17**
Qwen	**Greedy**	0.37					
	Nucleus	0.36					
		Fixed Weights			**Dynamic Weights**		
		Soft	Medium	Strong	Soft	Medium	Strong
	Greedy + UGLD-a	0.36	0.36	0.36	0.37	0.37	0.38
	Nucleus + UGLD-a	0.33	0.27	0.17	0.32	0.19	**0.12**

Table 4: In gray the best greedy configuration, in blue is the best nucleus configuration.

			Fleiss κ S1	Fleiss κ S2	UGLD Fluency %	Nucleus Fluency %	p-value
Anita	Paraphrasis	Soft	0.29	0.39	62	72	0.42
		Strong	0.37	0.54	56	54	1.0
	Free-text	Soft	0.2	0.3	82	78	0.80
		Strong	0.36	0.27	44	32	0.28
Qwen	Paraphrasis	Soft	0.43	0.45	68	76	0.57
		Strong	0.28	0.37	84	78	0.64
	Free-text	Soft	0.19	0.15	66	62	0.83
		Strong	0.33	0.20	62	60	1

Table 5: Results of the human evaluation.

tantly, agreement levels are similar across configurations, suggesting that UGLD does not introduce additional instability in perceived fluency. These findings seem to provide empirical support for our objective of developing a decoding-time conditioning approach that enhances lexical control without compromising fluency.

6. Conclusion

In this paper, we introduced Uncertainty-Gated Lexical Decoding (UGLD), a decoding-time framework for fine-grained lexical control in LLMs that explicitly addresses the trade-off between controllability and fluency. Unlike prior decoding-based control approaches, UGLD does not rely on additional training or external discriminators, making it particularly suitable for low-resource settings and languages. Instead, it leverages the entropy of the model's predictive distribution to adaptively scale lexical intervention, activating control when uncertainty is high and limiting interference when predictions are confident. This uncertainty-aware mechanism, combined with explicit lexical priors and logit-level penalties, constitutes the core novelty of our approach.

We evaluated the approach on two open-weight LLMs: ANITA, fine-tuned for Italian, and Qwen 3, a multilingual model supporting Italian generation. The twofold evaluation highlights complementary strengths of UGLD. On the one hand, automatic results show consistent improvements in lexical coverage, both in terms of increased use of simple vocabulary and reduced presence of expert-specific jargon in UGLD-based generations. The analysis further indicates that hyperparameter selection plays a crucial role: configurations that dynamically adapt the intervention at each decoding step and employ stronger intervention strengths produce the largest coverage shifts. On the other hand, human evaluation confirms that these lexical shifts do not negatively affect fluency. This is a particularly relevant outcome, as fluency degradation is a well-known limitation of decoding-time control approaches. Notably, annotators did not perceive a significant decline in fluency even under strong conditioning settings. Overall, these findings suggest that UGLD can serve as a reliable building block for future Controlled Text Simplification frameworks.

7. Acknowledgements

This work has been supported by the project "XAI-CARE" funded by the European Union - Next Generation EU - NRRP M6C2 "Investment 2.1 Enhancement and strengthening of biomedical research in the NHS" (PNRRMAD-2022-2376692_VADALA' – CUP F83C22002470001) and by LLMs4EU "Large

Language Models for the European Union" project, funded by the European Union through the Digital Europe Programme (DIGITAL-2024-AI-B-06-LANGUAGE - GA 101198470) under the grant agreement 101198470.
Partial support was also provided by the project "Understanding and Enhancing Preference Alignment in Large Language Models Through Controlled Text Generation" (IsCc8_ALIGNLLM), funded by CINECA under the ISCRA initiative, for HPC resource availability and support.

8. Lay Summary

Large Language Models (LLMs) are increasingly used to automatically generate text in a wide range of contexts. In many cases, however, it is important to control the type of language they produce. For instance, texts may need to use simpler and more accessible vocabulary, or avoid technical terminology when addressing non-expert readers. Achieving this type of control is challenging, as stronger constraints often reduce the fluency and naturalness of the generated text.

In this work, we propose a method that enables more precise control over the words used by these models while preserving the quality of the generated text. Our approach operates during the text generation process and does not require additional training resources. The key idea is to adjust the level of control depending on how uncertain the model is on what words to generate: stronger guidance is applied when the model is less certain, while more freedom is allowed when it is more confident.

We evaluate the method on Italian using two LLMs: ANITA, which is fine-tuned for Italian, and Qwen 3, a multilingual model capable of generating Italian text. The experiments focus on two scenarios: encouraging the use of simple vocabulary and reducing the presence of domain-specific jargon. In both cases, the method relies on external vocabularies, such as a list of common Italian words and a lexicon of technical terms, to guide the generation.

The results show that the proposed approach effectively increases the use of accessible words and reduces technical language, while maintaining fluency according to human evaluation. These findings suggest that the method can support applications aimed at improving the accessibility and readability of automatically generated texts.

9. Bibliographical References

Souvik Das, Lifeng Jin, Linfeng Song, Haitao Mi, Baolin Peng, and Dong Yu. 2025. Entropy guided extrapolative decoding to improve factuality in large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6589–6600, Abu Dhabi, UAE. Association for Computational Linguistics.

Tullio De Mauro and I Chiari. 2016. Il nuovo vocabolario di base della lingua italiana. *Internazionale*.

Aaron Grattafiori, Abhimanyu Dubey, and et al. 2024. The llama 3 herd of models.

Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1546, Vancouver, Canada. Association for Computational Linguistics.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. GeDi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online. Association for Computational Linguistics.

Clara Meister, Tiago Pimentel, Gian Wiher, and Ryan Cotterell. 2023. Locally typical sampling. *Transactions of the Association for Computational Linguistics*, 11:102–121.

Dang Nguyen, Jiuhai Chen, and Tianyi Zhou. 2024. Multi-objective linguistic control of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4336–4347, Bangkok, Thailand. Association for Computational Linguistics.

Damian Pascual, Beni Egressy, Clara Meister, Ryan Cotterell, and Roger Wattenhofer. 2021. A plug-and-play method for controlled text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3973–3997, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Marco Polignano, Pierpaolo Basile, and Giovanni Semeraro. 2024. Advanced natural-based interaction for the italian language: Llamantino-3-anita.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. 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 1314–1324, New Orleans, Louisiana. Association for Computational Linguistics.

Seungmin Shin, Dooyoung Kim, and Youngjoong Ko. 2025. ECO decoding: Entropy-based control for controllability and fluency in controllable dialogue generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 28297–28309, Suzhou, China. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025. Qwen3 technical report.

Kevin Yang and Dan Klein. 2021. FUDGE: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, Online. Association for Computational Linguistics.

Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. 2023. A survey of controllable text generation using transformer-based pre-trained language models. *ACM Comput. Surv.*, 56(3).

Tianqi Zhong, Quan Wang, Jingxuan Han, Yongdong Zhang, and Zhendong Mao. 2023. Air-decoding: Attribute distribution reconstruction for decoding-time controllable text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8233–8248, Singapore. Association for Computational Linguistics.

Wangchunshu Zhou, Yuchen Eleanor Jiang, Ethan Wilcox, Ryan Cotterell, and Mrinmaya Sachan. 2023. Controlled text generation with natural language instructions. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

A. Appendix

Table 6 shows an example in paraphrasing decoded with the Nucleus baseline and all the Nucleus+UGLD-t configurations. Words belonging to the New Basic Italian Vocabulary (NBIV) are highlighted in green.

As can be seen, the proportion of green words increases as the conditioning strength increases. For Anita, we can see that Anita Soft rewrites the sentence with the same number of green tokens as the Original, having a coverage lower than the Nucleus counterpart. For all the remaining Anita's settings and all Qwen's settings, higher coverage is achieved than in both the original sentence and the corresponding nucleus settings.

Curiously, while for Qwen the semantic content of the sentence is always correctly preserved during the paraphrasing, Anita either removes or changes the Proper noun of the sentence, from "Dario Arellano" to "Ennio Morricone". However, since this behavior also occurs in the baseline Nucleus setting, it appears to be model-specific rather than induced by UGLD.

Variant	Sentence (NVDB highlighted)	Coverage
Original	Figlio del noto produttore musicale Darío Arellano, fin da piccolo si dedica al teatro e alla recitazione. (lit. *Son of the well-known music producer Darío Arellano, from an early age he devotes himself to theatre and acting.*)	0.294
Nucleus Anita	Figlio del noto compositore musicale Ennio Morricone, da bambino si dedica al teatro e all'attoreplay. (lit. *Son of the well-known musical composer Ennio Morricone, as a child he devotes himself to theatre and to attoreplay*.)	0.312
Nucleus Qwen	Figlio del celebre musicista Darío Arellano, già da bambino si appassiona al teatro e alla recitazione. (lit. *Son of the celebrated musician Darío Arellano, already as a child he becomes passionate about theatre and acting.*)	0.188
Anita Soft	Figlio del noto compositore musicale Ennio Morricone, sin da bambino si dedica al teatro e alla recitazione. (lit. *Son of the well-known musical composer Ennio Morricone, from childhood he devotes himself to theatre and acting.*)	0.294
Anita Medium	Figlio del noto compositore musicale Ennio Morricone, fin da bambino si applica al teatro e all' attore . (lit. *Son of the well-known musical composer Ennio Morricone, from childhood he applies himself to theatre and to the actor.*)	**0.412**
Anita Strong	Figlio del famoso compositore musicale , fin da bambino si appassiona al palcoscenico e all' arte d' attore . (lit. *Son of the famous musical composer, from childhood he becomes passionate about the stage and the art of the actor.*)	**0.353**
Qwen Soft	Figlio del celebre musicista Darío Arellano, da giovanissimo si appassiona al teatro e alla recitazione. (lit. *Son of the celebrated musician Darío Arellano, at a very young age he becomes passionate about theatre and acting.*)	**0.200**
Qwen medium	Figlio del famosissimo musicista Darío Arellano da giovane si affaccia al teatro e alla recitazione. (lit. *Son of the very famous musician Darío Arellano, as a young man he approaches theatre and acting.*)	**0.267**
Qwen Strong	Figlio del celebre musicista Darío Arellano, fin da quando era un bambino si impegnava nel teatro e nello studio di recitazione. (lit. *Son of the celebrated musician Darío Arellano, from when he was a child he was engaged in theatre and in the study of acting.*)	**0.333**

Table 6: Example from Wikipedia in the paraphrasing scenario, generated with standard Nucleus sampling and with each Nucleus+UGLD-t settings. In **bold** all the settings that reach a higher coverage than its corresponding Nucleus baseline or Original, whichever is higher.