Autorégressivité dans l'espace latent d'un modèle de langage GP-VAE : une étude d'ablation empirique

Yves Ruffenach
Conservatoire National des Arts et Métiers
yves@ruffenach.net
ORCID: 0009-0009-4737-0555

Résumé

Les modèles de langage reposent presque exclusivement sur une factorisation autorégressive sur les tokens. Dans un travail antérieur, nous avons proposé une alternative dans laquelle la dynamique séquentielle est entièrement déplacée dans l'espace latent via un processus gaussien causal, tandis qu'un décodeur non-autorégressif opère en parallèle.

Nous menons une étude d'ablation systématique du rôle de l'autorégressivité dans l'espace latent. Nous comparons (i) la variante complète du modèle (GP-VAE AR), (ii) une ablation non-AR dans laquelle les latents deviennent indépendants, et (iii) un Transformer autorégressif sur les tokens.

Nos expériences montrent que, dans le régime étudié — c'est-à-dire avec des corpus de taille intermédiaire tels que WikiText-2 (~ 2 millions de tokens) et WikiText-103 (~ 100 millions de tokens) —, la suppression de l'autorégressivité dans l'espace latent réduit la log-densité moyenne sous le prior GP et conduit à un **effondrement systématique des générations longues** au-delà de 2048 tokens (dans le protocole expérimental considéré), malgré une perplexité locale encore correcte. À l'inverse, la variante AR produit des trajectoires latentes fortement corrélées, jugées plus probables par le prior gaussien, et permet de générer des séquences longues sans collapse dans ce même cadre expérimental, bien que les perplexités restent supérieures à celles de modèles Transformer plus optimisés.

Ces résultats doivent être lus comme une étude de faisabilité : ils fournissent des indices empiriques en faveur de l'idée que, dans ce cadre contrôlé, une partie de la séquentialité peut être portée par la dynamique latente même avec un décodeur non-autorégressif.

1 Introduction

Ce travail s'inscrit dans la continuité de l'article méthodologique précédent, intitulé *Modèle de langage GP-VAE à autorégressivité dans l'espace latent* [11], dans lequel est introduite et formalisée l'architecture GP-VAE à dynamique latente autorégressive. Le présent article constitue un complément, dédié à une étude d'ablation systématique. Il vise à quantifier le rôle exact de l'autorégressivité dans l'espace latent sur le comportement du GP-VAE.

Les modèles de langage contemporains reposent presque exclusivement sur une factorisation autorégressive dans l'espace des tokens, mise en œuvre par des architectures Transformer à attention multi-têtes. Cette approche atteint d'excellentes perplexités mais implique une génération strictement séquentielle et ne fournit pas de prior analytique explicite sur la dynamique temporelle : la structure de long terme est entièrement encodée dans les poids du réseau.

Des travaux récents ont exploré des modèles latents séquentiels dans lesquels la dépendance temporelle n'est plus portée par une récursion token-par-token, mais par une dynamique continue dans l'espace latent, notamment via des processus gaussiens causaux. Nous nous inscrivons dans cette ligne de recherche en étudiant un schéma de *latent autorégression* fondé sur un prior GP, couplé à un décodeur non-autorégressif parallèle. Ce paradigme vise à déplacer la séquentialité hors de l'espace des tokens pour la confier entièrement à une variable latente corrélée.

L'objectif de ce travail est expérimental : nous évaluons de manière systématique le rôle de l'autorégressivité dans l'espace latent à capacité fixe. Pour isoler son impact, nous comparons trois configurations strictement contrôlées :

1. un GP-VAE doté d'un latent autorégressif régi par un processus gaussien causal (AR) ;
2. une ablation non-AR où les latents deviennent indépendants tout en conservant les mêmes marges gaussiennes ;
3. un Transformer autorégressif sur les tokens.

Cette comparaison répond à trois questions fondamentales :

(Q1) Le modèle exploite-t-il réellement la structure corrélée imposée par le prior GP ?

(Q2) Que se passe-t-il lorsqu'on supprime explicitement la dynamique latente (ablation non-AR) ?

(Q3) Quelles propriétés séquentielles distinguent ce paradigme latent d'un modèle autorégressif de référence ?

Nos expériences montrent que la dynamique corrélée joue un rôle déterminant : la suppression de l'autorégressivité dans l'espace latent réduit la log-densité moyenne sous le prior GP et conduit à un effondrement systématique des générations à partir de 2048 tokens, malgré des perplexités internes encore raisonnables. À l'inverse, l'autorégressivité dans l'espace latent produit des trajectoires latentes fortement corrélées, jugées plus probables par le prior gaussien, et permet de générer des séquences longues sans collapse. Enfin, comparé à un Transformer dans ce même cadre expérimental, le modèle à autorégressivité dans l'espace latent présente un comportement séquentiel plus stable en génération longue, au sens des métriques de collapse définies précédemment, malgré une perplexité brute légèrement plus élevée.

Nous notons *GP-VAE AR* la variante à autorégressivité dans l'espace latent, et *GP-VAE non-AR* son ablation i.i.d.

Définition opérationnelle et périmètre des effondrements. Dans ce travail, le terme d'*effondrement* (ou *collapse*) désigne une dégradation générationnelle caractérisée par l'apparition de boucles déterministes ou quasi déterministes, mesurée opérationnellement par une fraction catastrophique cat_frac $= 1.0$, associée à des valeurs élevées de répétition (par exemple loop_frac ≈ 1 ou rep2, rep3 $\to 1$). Ces métriques sont définies en Section 4.3.

Les observations rapportées comme « systématiques » concernent l'ensemble des générations testées dans le cadre expérimental considéré, c'est-à-dire pour les prompts, checkpoints et seeds effectivement évalués dans nos expériences. Elles doivent être interprétées relativement à ce périmètre expérimental, et non comme une affirmation universelle sur le comportement du modèle au-delà de ces configurations.

En résumé, ce travail propose une *première évaluation empirique* du paradigme d'*autorégressivité purement latente* dans un régime caractérisé par des architectures légères, un contexte d'entraînement limité à $T_{\text{train}} = 64$ et deux corpus de référence : WikiText-2 et WikiText-103. Les observations que nous rapportons, cohérentes sur ces deux jeux de données de tailles et de variétés très différentes, suggèrent que, dans ce cadre particulier, la capacité séquentielle effective d'un modèle est fortement conditionnée par la dynamique latente plutôt que par la seule architecture du décodeur dans l'espace des tokens. Notre objectif consiste à documenter de manière contrôlée les effets de l'ablation AR vs non-AR au sein d'un même GP-VAE.

Afin de garantir la reproductibilité, le code utilisé pour l'ensemble des expériences est mis à disposition publiquement à l'adresse `https://github.com/y-v-e-s/GP-VAE-Latent-AR`.

2 Travaux connexes

Notre contribution se situe à l'intersection de trois lignes de travaux : les modèles de langage autorégressifs classiques, les modèles latents pour le texte, et les modèles séquentiels fondés sur des processus gaussiens.

2.1 Modèles de langage autorégressifs

Les modèles de langage dominants reposent sur une factorisation autorégressive de la forme

$$p(x_{1:T}) = \prod_{t=1}^{T} p(x_t \mid x_{<t}),$$

implémentée depuis [1] par des architectures Transformer. Ces modèles ont montré une capacité remarquable à approcher la distribution empirique du texte, avec des perplexités très faibles sur des corpus de grande taille, et constituent aujourd'hui la base des grands modèles de langage. Toutefois, cette expressivité se paie par une complexité d'inférence séquentielle (génération token par token) et par l'absence d'un prior analytique explicite sur la dynamique temporelle : la structure séquentielle est implicitement encodée dans les poids du réseau et dans la mécanique de l'attention, sans contrainte probabiliste globale sur les trajectoires.

Dans cette perspective, notre Transformer baseline joue le rôle d'un point de comparaison : un modèle autorégressif bien ajusté qui optimise la perplexité, mais qui peut souffrir de pathologies de génération (boucles et effondrements déterministes) en l'absence de régularisation séquentielle forte.

2.2 Modèles latents pour le texte

Les auto-encodeurs variationnels ont été étendus au domaine séquentiel pour introduire des variables latentes globales ou locales dans les modèles de langage [2, 3, 4]. Ces approches cherchent à combiner la flexibilité des décodeurs autorégressifs avec la capacité des latents à capturer des facteurs de variation de plus haut niveau (thème, style, discours). Un phénomène bien documenté est celui du "posterior collapse", où le décodeur AR absorbe toute l'information et rend le latent inutile, sauf si l'on impose des contraintes spécifiques (annealing de la KL, capacité limitée, structure hiérarchique, etc.).

Les variantes où le décodeur est non-autorégressif et où la dynamique séquentielle est portée par le latent sont plus rares, mais conceptuellement proches de notre travail. Elles posent la question de savoir si l'on peut déplacer entièrement la récursion dans l'espace latent, en laissant au décodeur un rôle

de mapping parallèle latent → tokens. Notre étude s'inscrit précisément dans ce cadre, en proposant une ablation contrôlée de la dynamique latente.

VAEs séquentiels : STORN, VRNN et SRNN. Au-delà des VAEs appliqués au texte [2], plusieurs modèles ont introduit des dynamiques latentes temporelles explicites. STORN [7] propose un modèle variationnel récurrent où l'état latent est mis à jour séquentiellement. VRNN [8] combine un RNN avec une variable latente par pas de temps, ce qui permet de capturer des variations stochastiques locales. SRNN [3] introduit quant à lui une factorisation latente hiérarchique distinguant un composant déterministe et un composant stochastique, améliorant la modélisation des séquences complexes. Ces travaux démontrent que l'introduction d'un latent séquentiel améliore la cohérence temporelle, mais reposent tous sur un décodeur autorégressif, contrairement à notre GP-VAE où la séquentialité est exclusivement portée par le latent.

Modèles de langage non-autorégressifs (NAT). Les modèles non-autorégressifs ont été introduits principalement dans le contexte de la traduction [9, 10], où ils permettent une génération parallèle en remplaçant la factorisation séquentielle $p(x_t \mid x_{<t})$ par une prédiction indépendante ou itérative. Ces modèles sacrifient souvent la qualité locale pour obtenir un accélération de génération. Notre approche diffère fondamentalement : le décodeur est effectivement non-autorégressif, mais la structure séquentielle n'est pas supprimée. Elle est transférée intégralement dans l'espace latent via un processus gaussien causal, ce qui permet de conserver une cohérence temporelle globale tout en bénéficiant d'un décodeur parallèle.

2.3 Processus gaussiens et modèles séquentiels latents

Les processus gaussiens (GP) sont traditionnellement utilisés comme modèles non paramétriques pour l'approximation de fonctions et les séries temporelles [5]. Leur utilisation comme priors analytiques dans des VAEs séquentiels a été explorée dans divers contextes (trajectoires continues, séries temporelles, données spatio-temporelles), souvent sous la forme de GP-VAE où les latents suivent une loi gaussienne corrélée [6].

Dans notre travail méthodologique antérieur, nous avons introduit un GP-VAE pour le langage dans lequel la dynamique séquentielle est entièrement portée par un processus gaussien causal sur le latent, tandis que le décodeur reste parallèle. Le présent article se concentre sur l'aspect empirique

de ce schéma : nous ne modifions pas le modèle, mais étudions systématiquement l'impact de la composante autorégressive latente (GP causal) en la comparant à une ablation non-AR et à un Transformer autorégressif sur les tokens. L'objectif est de quantifier dans quelle mesure le prior GP est effectivement exploité, et si la dynamique latente peut jouer un rôle comparable à l'autorégressivité symbolique des Transformers.

3 Rappel du modèle GP-VAE à autorégressivité dans l'espace latent

Nous considérons le GP-VAE introduit dans le travail méthodologique précédent, dans lequel une séquence de tokens $x_{1:T}$ est encodée par un réseau de convolutions dilatées causales (TCN) en une trajectoire latente $z_{1:T} \in \mathbb{R}^{T \times d_z}$. Un décodeur non-autorégressif opère ensuite en parallèle sur l'ensemble de la trajectoire latente pour produire une distribution sur les tokens, ce qui permet de dissocier complètement la dynamique temporelle (portée par le latent) et la génération symbolique.

Factorisation et parallélisme du décodeur. Le décodeur opère de manière entièrement parallèle sur la trajectoire latente complète $z_{1:T}$. Concrètement, un réseau convolutionnel causal f_θ prend en entrée $z_{1:T}$ et produit en une seule passe les logits sur tous les tokens,

$$(\ell_1, \dots, \ell_T) = f_\theta(z_{1:T}), \qquad \ell_t \in \mathbb{R}^{|\mathcal{V}|}, \tag{1}$$

où $\mathcal{V}$ désigne le vocabulaire.

Conditionnellement au latent, la distribution générative sur les tokens factorise donc *par position* :

$$p_\theta(x_{1:T} \mid z_{1:T}) = \prod_{t=1}^{T} p_\theta(x_t \mid z_{1:T}) = \prod_{t=1}^{T} \mathrm{Cat}\big(x_t \mid \mathrm{softmax}(\ell_t)\big). \tag{2}$$

Les dépendances séquentielles entre tokens sont ainsi portées uniquement par la structure temporelle du latent (via le prior GP et le TCN encodeur), et non par une factorisation autorégressive explicite dans l'espace des tokens.

Dans la variante *non-AR*, chaque vecteur latent est traité comme une variable conditionnellement indépendante, dépendant uniquement de l'observation locale : le processus latent ne possède pas de structure temporelle propre, et le prior (gaussien complet ou diagonal) n'impose qu'une contrainte

ponctuelle sur la norme des vecteurs. Dans ce régime, le GP-VAE se rapproche d'un auto-encodeur variationnel bruité avec un couplage temporel très limité.

À l'inverse, dès lors que l'on impose un lien autorégressif entre les latents successifs, ceux-ci cessent d'être de simples "codes" indépendants pour devenir les réalisations discrétisées d'un processus stochastique continu, en l'occurrence un processus gaussien temporel. Le modèle devient alors un véritable modèle séquentiel latent, dans lequel la structure temporelle est entièrement portée par la loi a priori.

Le rôle du prior GP devient alors central : il ne se contente plus de régulariser la magnitude des latents, il contraint directement la géométrie de leurs trajectoires (lissage, corrélation à court terme, continuité). L'apprentissage doit simultanément satisfaire la reconstruction et respecter cette structure temporelle, ce qui transforme le GP-VAE en un modèle séquentiel latent au sens fort. En revanche, sans composante autorégressive, le modèle peut atteindre une perplexité correcte tout en violant complètement les hypothèses du GP, ce que révèlent immédiatement les log-densité moyennes sous le prior GP et les statistiques d'auto-corrélation.

Objectif variationnel. Pour une séquence $x_{1:T}$, le GP-VAE est entraîné par maximisation de l'ELBO standard

$$\mathcal{L}(x_{1:T}) = \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})}\big[\log p_\theta(x_{1:T} \mid z_{1:T})\big] - \mathrm{KL}\big(q_\phi(z_{1:T} \mid x_{1:T}) \,\|\, p_\theta(z_{1:T})\big), \tag{3}$$

où $q_\phi(z_{1:T} \mid x_{1:T})$ est un posterior gaussien diagonal temporel et $p_\theta(z_{1:T})$ un prior gaussien corrélé.

Dans le régime AR, le prior latent est factorisé de façon causale

$$p_\theta(z_{1:T}) = \prod_{t=1}^{T} p_\theta(z_t \mid z_{<t}) = \prod_{t=1}^{T} \mathcal{N}\big(z_t \mid \mu_{t|<t}, \Sigma_{t|<t}\big), \tag{4}$$

où $(\mu_{t|<t}, \Sigma_{t|<t})$ sont obtenus par mise à jour gaussienne explicite à partir du kernel du processus gaussien.

Dans l'ablation *non-AR*, on remplace ce prior corrélé par un prior i.i.d. de mêmes variances marginales

$$p_\theta^{\text{non-AR}}(z_{1:T}) = \prod_{t=1}^{T} \mathcal{N}\big(z_t \mid 0, \mathrm{diag}(K_{tt})\big), \tag{5}$$

de sorte que toute corrélation temporelle dans le latent est supprimée.

Dans cette perspective, l'ablation non-AR fournit un test direct : si le modèle exploite réellement la dynamique corrélée imposée par le processus

gaussien, supprimer l'autorégression latente doit se traduire à la fois par (i) une chute drastique de compatibilité avec le prior et (ii) une dégradation des métriques de génération textuelle.

4 Protocole expérimental

Implémentations indépendantes du GP-VAE

Deux implémentations indépendantes du GP-VAE ont été développées et utilisées dans cette étude afin de dissocier les effets attribués à la dynamique latente des choix d'architecture de l'encodeur et des détails d'implémentation.

- une implémentation dite *pyramidale*, fondée sur un encodeur TCN hiérarchique à décroissance de résolution temporelle ;
- une implémentation dite *TCN+*, reposant sur un réseau convolutionnel temporel dilaté plus profond, associée à une implémentation vectorisée du prior GP.

Ces deux implémentations partagent exactement le même espace latent, le même kernel de processus gaussien, le même prior GP au sens mathématique, le même objectif variationnel (ELBO) et le même décodeur non-autorégressif parallèle. Elles ne diffèrent que par l'architecture de l'encodeur et par l'implémentation computationnelle du prior (séquentielle vs vectorisée).

Les deux implémentations sont utilisées au cours de l'article, parfois sur un même corpus, avec des rôles distincts (résultats principaux, réplications ou analyses de robustesse). Chaque section précise explicitement quelle implémentation est employée.

(1) Implémentation Pyramidale–GP-VAE. Encodeur TCN pyramidal fondé sur des convolutions dilatées causales à réceptif croissant ; génération latente séquentielle via les conditionnelles exactes du processus gaussien causal $p(z_t \mid z_{<t})$.

(2) Implémentation TCN+–GP-VAE. Variante plus récente utilisant un encodeur TCN étendu ("TCN+"), avec une formulation vectorisée du prior GP, optimisée pour les batchs volumineux et les corpus de grande échelle (notamment WikiText-103).

Ces deux implémentations permettent de vérifier que les phénomènes observés — notamment l'écart AR vs non-AR dans l'espace latent — ne proviennent pas d'un artefact architectural mais sont reproductibles dans deux cadres indépendants.

Lorsque plusieurs implémentations sont utilisées sur un même corpus, l'implémentation de référence est explicitement indiquée dans la section correspondante. À titre indicatif, l'implémentation TCN+ s'est révélée empiriquement plus stable dans plusieurs configurations, tandis que l'implémentation pyramidale est utilisée comme réplication et contrôle indépendant. Les observations clés (compatibilité avec le prior, structure temporelle latente, métriques de génération) ont été répliquées avec l'implémentation pyramidale, ce qui confirme leur robustesse.

	Implémentation Pyramidal–GP-VAE	**Implémentation TCN+–GP-VAE**
Encodeur	TCN pyramidal (convolutions dilatées causales à réceptif croissant)	TCN+ : TCN étendu, plus profond, optimisé pour des corpus volumineux
Génération latente	Conditionnelles exactes $p(z_t \mid z_{<t})$ calculées pas à pas	Formulation vectorisée / optimisée du GP (batchs volumineux)
Prior GP	GP causal stationnaire, mis à jour séquentiellement	Même GP, implémentation vectorisée plus efficace
Décodeur	Identique : CNN non-autorégressif parallèle	Identique
Objectif ELBO	Identique (reconstruction + KL cap)	Identique
Usage dans l'étude	Utilisée sur WikiText-2 et WikiText-103 (réplication des tendances AR vs non-AR)	Utilisée sur WikiText-2 et WikiText-103 (résultats principaux)
Rôle dans l'étude	Réplication et contrôle indépendant	Résultats principaux et expérimentations à grande échelle

Table 1 – Comparaison des deux implémentations indépendantes du GP-VAE utilisées dans l'étude.

4.1 Variantes étudiées

Nous étudions deux variantes d'un même GP-VAE, qui ne diffèrent que par la dynamique dans l'espace latent. L'encodeur, le décodeur et l'objectif ELBO sont strictement identiques, seule la forme du prior latent varie.

GP-VAE à autorégressivité dans l'espace latent (AR). La trajectoire latente $z_{1:T}$ est générée séquentiellement via un processus gaussien causal,

$$p(z_{1:T}) = \prod_{t=1}^{T} p(z_t \mid z_{<t}),$$

où chaque conditionnelle $p(z_t \mid z_{<t})$ est obtenue par mise à jour gaussienne explicite à partir du kernel du processus gaussien. Cette factorisation réalise un processus gaussien causal sur la grille temporelle discrète.

GP-VAE non-AR (ablation). La même covariance GP est utilisée pour définir les variances marginales, mais les latents sont échantillonnés de manière indépendante,

$$z_t \sim \mathcal{N}\big(0, \mathrm{diag}(K_{tt})\big),$$

de sorte que toute corrélation temporelle est supprimée. Les deux variantes partagent le même encodeur TCN, le même décodeur non-autorégressif et le même objectif ELBO ; seule la dynamique de génération dans le latent est modifiée.

Baseline Transformer autorégressif sur les tokens. En complément, nous entraînons un Transformer autorégressif sur les tokens, utilisé comme baseline de comportement pour situer la qualité linguistique et la stabilité des continuations produites par le GP-VAE, sans rechercher de performance état de l'art.

4.2 Corpus, modèles de référence et configuration d'entraînement

Les expériences principales sont menées sur le corpus WikiText-2 (version *raw*). Le texte est tokenisé avec le tokenizer GPT-2 (fichiers `tokenizer.json`, `vocab.json`, `merges.txt`), puis concaténé et découpé en blocs de longueur fixe, avec une longueur de contexte d'entraînement fixée à $T_{\text{train}} = 64$ tokens pour les logs reportés. Nous utilisons les splits standard train/validation/test, sans fuite d'information entre splits.

Longueurs de contexte et de génération. Dans tout l'article, nous distinguons explicitement :

- T_{train}, la longueur de contexte utilisée lors de l'entraînement, correspondant à la taille des blocs (block size) sur lesquels l'ELBO est optimisée ;
- L_{gen}, la longueur des séquences générées ou évaluées lors des expériences de continuation, pouvant excéder largement la longueur de contexte d'entraînement.

Sauf mention contraire, les modèles sont entraînés avec une longueur de contexte fixée à $T_{\text{train}} = 64$, tandis que les expériences de génération et

d'évaluation explorent des longueurs de continuation L_{gen} variables, allant de quelques dizaines à plusieurs milliers de tokens.

Architecture du GP-VAE. L'encodeur est un réseau de convolutions dilatées causales (TCN) projetant les tokens en une séquence latente de dimension d_z. Le posterior est un gaussien diagonal temporel

$$q_\phi(z_{1:T_{\text{train}}} \mid x_{1:T_{\text{train}}}).$$

Le prior latent est un processus gaussien stationnaire

$$z_{1:T_{\text{train}}} \sim \mathcal{N}(0, K \otimes I_{d_z}),$$

et le décodeur est un réseau convolutionnel léger appliqué en parallèle à la trajectoire latente complète, produisant des logits de tokens conditionnés sur $z_{1:T_{\text{train}}}$.

Objectif et optimisation. L'entraînement se fait par maximisation de l'ELBO moyen par token, avec :

- un terme de reconstruction basé sur une cross-entropie lissée (label smoothing) ;
- un terme de divergence de Kullback–Leibler entre un posterior diagonal temporel et le prior GP corrélé ;
- une régularisation supplémentaire sur les embeddings de sortie.

La divergence KL est plafonnée par un paramètre `kl_cap` et pondérée par un coefficient β adapté dynamiquement de manière à maintenir la KL/token autour d'une cible prédéfinie. Sur WikiText-2, nous fixons `kl_cap` $= 8$ nats et faisons croître β jusqu'à environ 0,35 en fin d'entraînement.

Nous fixons la valeur du *KL cap* à 8 pour l'ensemble des expériences principales. Afin de vérifier que ce choix n'introduit pas d'effet de paramétrage, nous avons mené une brève analyse de sensibilité en évaluant les variantes $\{4, 8, 16\}$. Toutes présentent un comportement qualitatif identique (mêmes tendances AR vs non-AR, mêmes phénomènes de collapse ou de stabilité, mêmes ordres de grandeur de log-densité moyenne sous le prior GP). Pour la lisibilité et l'homogénéité des résultats, nous reportons la configuration avec *KL cap* $= 8$.

4.3 Métriques d'évaluation

Nous utilisons trois familles de métriques : latentes, textuelles internes et textuelles externes (sous GPT-2).

Terminologie. Dans la suite, nous utilisons le terme *log-vraisemblance* (LL) pour désigner les quantités associées à la vraisemblance du décodeur conditionnel (par ex. LL_0, $\mathrm{LL}_{\mathrm{multi}}$). En revanche, pour les distributions a priori continues sur les trajectoires latentes, nous parlons de *log-densité* (par ex. $\log p_{\mathrm{GP}}(z)$), afin d'éviter toute ambiguïté liée à l'usage du terme « probabilité » pour des lois à densité.

Métriques latentes.

- log-densité moyenne $\log p_{\mathrm{GP}}(z)$ des trajectoires sous le prior gaussien corrélé.
- log-densité moyenne $\log p_{\mathrm{diag}}(z)$ sous un prior diagonal i.i.d. de même variance marginale.
- Auto-corrélation cosinus moyenne corr_k pour des lags $k = 1, \ldots, 10$.
- Norme moyenne des pas latents $\|z_t - z_{t-1}\|$.

Métriques textuelles internes.

Perplexité conditionnelle du GP-VAE. La perplexité reportée pour le GP-VAE correspond à la *cross-entropie conditionnelle* du décodeur non-autorégressif :

$$p_\theta(x_{1:T} \mid z_{1:T}) = \prod_{t=1}^{T} p_\theta(x_t \mid z_{1:T}).$$

Elle mesure uniquement la qualité locale du décodage conditionné sur la trajectoire latente complète. Il ne s'agit **pas** d'une perplexité de modèle de langage autorégressif, qui reposerait sur la factorisation :

$$p(x_{1:T}) = \prod_{t=1}^{T} p(x_t \mid x_{<t}).$$

Par conséquent :

- les valeurs de PPL du GP-VAE **ne sont pas comparables numériquement** avec celles d'un Transformer ou d'un GPT-2 ;
- des valeurs très faibles (par ex. 2–4) sont possibles, car le modèle conditionnel dispose d'une information latente globale ;
- cette perplexité doit être interprétée uniquement comme une mesure interne de cohérence du décodeur conditionnel.
- Perplexité en validation PPL(val).
- Perplexité de continuation PPL(cont) sur un protocole prompt + complétion en *teacher forcing*.

— Taux de répétitions (bigrams/trigrams répétés), fraction de tokens appartenant à une boucle exacte et fréquence de séquences catastrophiques.

Remarque méthodologique sur la comparabilité des perplexités. Toutes les comparaisons de perplexité rapportées dans ce travail doivent être lues *dans la factorisation probabiliste propre à chaque modèle*, et non comme une comparaison directe de modèles autorégressifs sur les tokens. Les perplexités du GP-VAE correspondent à des métriques internes conditionnelles, tandis que celles des Transformers reposent sur une factorisation strictement autorégressive sur les tokens.

Métriques via GPT-2 (juge externe).

— NLL et perplexité de la continuation sous GPT-2, conditionnée sur le même prompt.
— Fraction de tokens « rares » (probabilité GPT-2 $< 10^{-4}$).
— Auto-similarité moyenne et maximale de fenêtres de taille fixe dans les embeddings GPT-2.

4.4 Positionnement et limites des baselines

Avant de présenter les résultats, il est important de préciser le statut des modèles de référence.

Transformer autorégressif sur les tokens. Le Transformer autorégressif sur les tokens « minimal » que nous considérons poursuit avant tout un objectif de référence contrôlée, sans vocation à constituer un état de l'art compétitif sur WikiText-2 ou WikiText-103. Il s'agit d'un modèle à capacité contrôlée, entraîné avec un budget d'itérations et un espace d'hyperparamètres volontairement restreints, afin de servir de point de comparaison comportemental plutôt que de baseline optimisée.

GPT-2 comme baseline et juge externe. GPT-2 intervient sous deux formes distinctes : (i) un GPT-2 pré-entraîné, gelé, utilisé exclusivement comme *juge externe* pour mesurer la plausibilité linguistique des continuations (perplexité externe, tokens rares, similarité d'embeddings) ; (ii) un GPT-2 légèrement fine-tuné sur WikiText-2, utilisé comme baseline supplémentaire. Dans les deux cas, ces modèles sont plus massifs et ont bénéficié d'un pré-entraînement sur des corpus auxquels nos GP-VAE n'ont pas accès. Les

comparaisons de perplexité s'inscrivent dans une démarche de repérage qualitatif plutôt que dans une logique de revendication de performance compétitive au sens large, et fournissent avant tout des points d'ancrage pour situer le comportement des variantes latentes.

L'objectif principal de ce travail est d'analyser, *à capacité fixe et architecture donnée*, l'impact de l'autorégressivité purement latente à l'intérieur d'un même GP-VAE (AR vs non-AR), puis de confronter ce paradigme à un Transformer dans le but de mettre en évidence des phénomènes de stabilité et de collapse. Le présent travail vise avant tout l'étude du rôle de la dynamique latente dans un cadre expérimental contrôlé, avec un positionnement distinct de toute recherche de dépassement des Transformers modernes bien entraînés.

5 Résultats sur WikiText-2

5.1 Implémentations utilisées sur WikiText-2

Les expériences rapportées sur WikiText-2 utilisent les deux implémentations indépendantes du GP-VAE décrites précédemment.

Sauf mention explicite contraire, les résultats détaillés dans cette section correspondent à l'implémentation pyramidale, qui constitue notre implémentation historique et la première version validée expérimentalement du modèle. Les principales tendances (écart AR vs non-AR, structure latente, stabilité en génération longue) ont également été vérifiées avec l'implémentation TCN+, afin d'en confirmer la robustesse vis-à-vis de l'architecture de l'encodeur et des choix d'implémentation du prior.

Afin de clarifier la lecture, nous organisons cette section en six volets complémentaires : (i) la mécanique latente (normes, corrélations, compatibilité avec le prior GP), (ii) la perplexité interne et la dynamique d'entraînement, (iii) la génération courte, (iv) la génération longue, (v) l'évaluation externe par un juge GPT-2, et (vi) le contraste « collapse vs stabilité » entre les différentes variantes. Les tableaux et sous-sections suivants suivent explicitement cette structure.

5.2 Implémentation pyramidale utilisée

Sauf mention explicite contraire, l'ensemble des résultats présentés dans cette section s'appuie sur notre implémentation originale du GP-VAE — que nous noterons *implémentation pyramidale*. Cette version utilise un encodeur TCN pyramidal fondé sur des convolutions dilatées causales, qui paramètre un posterior gaussien diagonal $q_\phi(z_{1:T} \mid x_{1:T})$. Le prior latent est un processus

gaussien stationnaire $z_{1:T} \sim \mathcal{N}(0, K \otimes I_{d_z})$, et la génération autorégressive est implémentée explicitement pas à pas : chaque z_t est échantillonné selon la conditionnelle exacte $p(z_t \mid z_{<t})$, obtenue à partir de la structure du GP. Cette implémentation inclut également une ablation « non-AR » obtenue en remplaçant K par $\mathrm{diag}(K)$, ce qui supprime les dépendances temporelles dans l'espace latent. Les résultats présentés dans cette section concernent exclusivement cette implémentation.

5.3 Dynamique d'entraînement

Le modèle GP-VAE à autorégressivité dans l'espace latent est entraîné sur WikiText-2 raw, avec des blocs de longueur $T_{\text{train}} = 64$. La perplexité d'entraînement chute très rapidement, puis se stabilise autour de 3–4, tandis que le terme KL/token plafonné reste au cap fixé (8 nats) et que le coefficient β est progressivement augmenté jusqu'à environ 0,35.

Quelques points caractéristiques de la courbe d'entraînement sont résumés dans le Tableau 2.

Step	ELBO/tok	LL_0/tok	$\mathrm{LL}_{\text{multi}}$/tok	KL_{cap}/tok	PPL(train)
50	-14.79	-8.94	-5.56	8.0	7438.45
400	-8.33	-2.24	-5.37	8.0	8.34
1000	-7.64	-1.46	-4.68	8.0	3.76
1500	-8.10	-1.48	-4.47	8.0	3.85
2000	-8.65	-1.41	-4.42	8.0	3.59

Table 2 – Évolution de l'ELBO/token, des termes de log-vraisemblance et de la perplexité d'entraînement sur WikiText-2 pour le GP-VAE à autorégressivité dans l'espace latent.

Remarque. La perplexité reportée ici est une perplexité **interne conditionnelle** du GP-VAE (voir Section 4.3) et non une perplexité au sens autorégressif. Elle ne doit donc **pas** être comparée directement aux perplexités obtenues par les modèles Transformer.

La validation finale donne :

$$\begin{aligned} \text{ELBO/token} &= -8.266, \\ \text{LL}_0 &= -0.953, \\ \text{LL}_{\text{multi}} &= -4.363, \\ \text{KL/token} &= 8.0, \\ \text{PPL(val)} &= 2.25. \end{aligned}$$

Sur WikiText-2, le GP-VAE atteint ainsi une perplexité de validation de 2,25, avec un ELBO/token de $-8{,}27$ et un terme KL/token plafonné à 8 nats. La perplexité d'entraînement se stabilise autour de 3–4, ce qui indique un léger sur-apprentissage mais une généralisation encore correcte sur l'ensemble de validation.

5.4 Statistiques dans l'espace latent : AR vs non-AR

5.4.1 Dynamique des pas latents

Les statistiques suivantes résument la dynamique des pas latents pour les deux régimes (AR vs non-AR) : AR : step_norm_mean = 4.8678 (std = 0.9826), cos_mean = 0.5661 (std = 0.1156) ; non-AR : step_norm_mean = 8.4568 (std = 0.7381), cos_mean = -0.0008 (std = 0.1242).

Ces valeurs sont reportées dans le Tableau 3.

Régime latent	$\mathbb{E}\|z_t - z_{t-1}\|$	$\mathbb{E}[\cos(z_t, z_{t-1})]$
AR	4.87 ± 0.98	0.57 ± 0.12
non-AR	8.46 ± 0.74	$\approx 0.00 \pm 0.12$

Table 3 – Dynamique des pas latents sur WikiText-2 : amplitude moyenne et corrélation cosinus entre pas successifs.

L'autorégressivité dans l'espace latent produit des trajectoires latentes plus lisses (pas environ deux fois plus petits en norme) et fortement corrélées d'un pas au suivant (cosinus moyen $\approx 0{,}57$), alors que la variante non-AR se comporte comme un bruit blanc (cosinus moyen ≈ 0).

5.4.2 Compatibilité avec le prior GP vs prior diagonal

Les log-densité moyennes des trajectoires latentes sous le prior GP corrélé et sous un prior diagonal i.i.d. de même variance marginale sont les suivantes

:

$$\text{AR} : \log p_{\text{GP}}(z) = 2313.35, \quad \log p_{\text{diag}}(z) = -4299.85,$$
$$\text{non-AR} : \log p_{\text{GP}}(z) = -26{,}632{,}656.00, \quad \log p_{\text{diag}}(z) = -4668.72.$$

Régime latent	$\log p_{\text{GP}}(z)$	$\log p_{\text{diag}}(z)$
AR	$+2.31 \times 10^3$	-4.30×10^3
non-AR	-2.66×10^7	-4.67×10^3

TABLE 4 – log-densité moyennes des trajectoires latentes sous le prior GP corrélé et sous un prior diagonal i.i.d. de même variance marginale sur WIKITEXT-2.

Sous WIKITEXT-2, les trajectoires AR sont jugées plus probables par le prior GP ($\log p_{\text{GP}} \approx 2{,}3 \times 10^3$), alors que les trajectoires non-AR sont rejetées ($\log p_{\text{GP}} \approx -2{,}7 \times 10^7$). **Autrement dit, l'ablation non-AR réduit la log-densité moyenne latente sous le prior GP, transformant un latent séquentiel compatible en un bruit que le GP rejette quasi systématiquement.** À l'inverse, les deux régimes obtiennent des log-densités comparables sous un prior diagonal, ce qui montre que la différence ne vient pas de la norme des vecteurs mais bien de la structure temporelle imposée par le GP.

Un avantage du prior GP qui ne relève pas d'un artefact d'optimisation. On pourrait objecter que la meilleure compatibilité de l'autorégressivité dans l'espace latent avec le prior gaussien est en partie « câblée » par l'objectif d'apprentissage, puisque la divergence KL est elle-même calculée par rapport à ce prior. Deux éléments viennent nuancer cette objection. Premièrement, les deux variantes partagent exactement le même objectif et le même cap de KL/token : elles sont soumises à la même pression de régularisation vers le GP, et la KL/token atteint dans les deux cas la valeur plafond imposée. Si le prior était « gagné d'avance », on s'attendrait à des log-densité moyennes comparables sous p_{GP} pour AR et non-AR, ce qui n'est pas le cas : la variante non-AR est rejetée par le prior alors même qu'elle respecte numériquement la contrainte de KL.

Deuxièmement, nous comparons systématiquement $p_{\text{GP}}(z)$ à un prior diagonal i.i.d. ayant la même variance marginale. Le fait que les deux régimes obtiennent des log-densités similaires sous ce prior diagonal, tout en divergeant fortement sous le prior corrélé, indique que la différence ne porte pas sur

l'échelle des latents mais bien sur leur structure temporelle. Autrement dit, le modèle AR exploite effectivement la dynamique corrélée permise par le GP, là où le modèle non-AR reste dans un régime de codes quasi indépendants malgré une KL/token saturée.

5.4.3 auto-corrélation multi-lag

L'auto-corrélation cosinus moyenne en fonction du lag k donne le profil suivant :

Lag k	AR	non-AR
1	0.567	-0.000
2	-0.287	0.001
3	-0.730	-0.000
4	-0.382	-0.001
5	0.313	-0.000
6	0.621	-0.001
7	0.269	-0.003
8	-0.316	0.000
9	-0.525	0.001
10	-0.185	0.001

L'auto-corrélation cosinus des latents AR présente un profil oscillant marqué (par exemple 0,57 au lag 1, $-0{,}29$ au lag 2, $-0{,}73$ au lag 3), révélant une dépendance temporelle non triviale incompatible avec un bruit i.i.d.

Interprétation L'auto-corrélation mesurée ici correspond à une *corrélation cosinus* entre les vecteurs latents z_t et z_{t+k}, et non à une corrélation linéaire scalaire au sens classique des processus gaussiens. Le profil observé pour la variante AR présente des alternances de signe (positif / négatif) selon le lag k, ce qui reflète principalement des *changements de direction moyenne* dans l'espace latent plutôt qu'une oscillation explicite du processus sous-jacent.

De telles alternances peuvent apparaître même lorsque le prior GP est défini par un kernel lisse (par exemple de type RBF), dès lors que l'on considère une corrélation angulaire normalisée dans un espace de dimension finie. Elles traduisent une structure temporelle non triviale des trajectoires latentes, incompatible avec un bruit i.i.d., mais ne doivent pas être interprétées comme la signature directe d'un kernel oscillant.

À l'inverse, la variante non-AR présente une corrélation cosinus systématiquement proche de zéro pour tous les lags, ce qui confirme l'absence de structure temporelle dans l'espace latent.

5.5 Qualité de génération sous GPT-2

5.5.1 Perplexité et tokens rares

Les métriques de génération sont évaluées avec un modèle GPT-2 externe comme juge, pour des continuations de longueur $L \in \{32, 64, 128, 256, 512\}$. Les perplexités GPT-2 et fractions de tokens rares (probabilité $< 10^{-4}$) sont résumées dans le Tableau 5.

	AR latent		non-AR latent	
L	PPL(GPT-2)	rare_frac	PPL(GPT-2)	rare_frac
32	3.79×10^3	0.327	2.09×10^5	0.760
64	3.80×10^3	0.343	1.41×10^5	0.792
128	7.86×10^3	0.506	9.98×10^4	0.789
256	1.92×10^4	0.641	5.29×10^4	0.746
512	2.55×10^4	0.693	3.04×10^4	0.717

TABLE 5 – Perplexité GPT-2 et fraction de tokens rares (proba $< 10^{-4}$) pour des continuations de longueur L générées par le GP-VAE (AR vs non-AR) sur WIKITEXT-2.

Pour toutes les longueurs jusqu'à $L = 256$, la PPL GPT-2 est plus faible pour AR que pour non-AR (parfois d'un facteur de 5 à 50). La fraction de tokens rares est systématiquement plus basse pour AR (0,33–0,69) que pour non-AR (0,74–0,79). À $L = 512$, l'écart de PPL se réduit, mais AR reste légèrement meilleur en rare_frac (0,693 vs 0,717). Ces observations indiquent que les séquences AR restent plus proches de la distribution linguistique cible.

5.5.2 Répétitions et « looping »

Les métriques de répétition (bigrammes/trigrammes répétés, répétitions consécutives, boucles exactes) donnent le profil suivant :

L	Régime	rep2	rep3	consec	loop_frac
32	AR	0.259	0.087	0.214	0.031
	non-AR	0.007	0.000	0.027	0.000
64	AR	0.311	0.141	0.201	0.012
	non-AR	0.011	0.002	0.034	0.004
128	AR	0.203	0.074	0.132	0.007
	non-AR	0.011	0.000	0.032	0.001
256	AR	0.121	0.036	0.093	0.003
	non-AR	0.039	0.005	0.047	0.005
512	AR	0.093	0.025	0.077	0.006
	non-AR	0.067	0.017	0.063	0.015

L'ablation non-AR produit des séquences globalement moins répétitives (rep2 et rep3 plus faibles), mais au prix d'une perplexité GPT-2 élevée et d'un taux de tokens rares supérieur. L'autorégressivité dans l'espace latent introduit davantage de répétitions locales, sans toutefois tomber dans des boucles catastrophiques (loop_frac et cat_frac ≈ 0), ce qui suggère une meilleure cohérence locale plutôt qu'un « mode collapse » trivial.

Régime non-AR. À court horizon ($L \leq 256$), la variante non-AR est moins plausible sous GPT-2 : les perplexités externes sont beaucoup plus élevées que pour l'autorégressivité dans l'espace latent (parfois d'un facteur 10 à 50) et la fraction de tokens rares plus importante. Les séquences restent toutefois exemptes de répétitions à ces longueurs, ce qui reflète l'absence de structure récurrente stable dans le latent. Cependant, cette absence totale de corrélation rend le modèle fragile : lorsque l'on augmente l'horizon de génération, à partir de $L = 2048$, puis systématiquement à $L = 3072$, toutes les générations non-AR s'effondrent en mode *collapse*, avec une fraction catastrophique cat_frac $= 1.0$.

Régime AR. La variante AR présente des taux de répétitions plus élevés, ce qui suggère une plus forte cohérence locale induite par le processus gaussien. Toutefois, contrairement au régime non-AR, elle reste parfaitement stable même pour des séquences très longues : aucune séquence catastrophique n'est observée jusqu'à $L = 3072$. La perplexité GPT-2 reste plus élevée qu'en non-AR, mais les continuations demeurent linguistiquement valides et ne dérivent pas vers des boucles infinies.

En résumé, sur WikiText-2, l'autorégressivité dans l'espace latent est à la fois jugée plus probable par GPT-2 à court horizon (perplexité externe

plus faible, moins de tokens rares) et plus stable à longue portée, tandis que le régime non-AR reste hors distribution et s'effondre systématiquement pour des séquences très longues.

5.6 Évaluation externe ponctuelle par GPT-2 sur WikiText-2

Nous commençons par une expérience contrôlée sur WikiText-2, fondée sur un prompt court « The meaning of life » suivi d'une continuation de longueur 64 tokens. Le GP-VAE à autorégressivité dans l'espace latent est entraîné avec des blocs de longueur $T_{\text{train}} = 64$ et atteint en validation

$$\begin{aligned} \text{ELBO/token} &= -6.704, \\ \text{LL}_0 &= -1.027, \\ \text{LL}_{\text{multi}} &= -4.805, \\ \text{KL/token} &= 8.0, \\ \text{PPL(val)} &= 2.43. \end{aligned}$$

Si l'on ré-évalue la continuation générée par le GP-VAE AR sous le même modèle (teacher forcing sur la partie continuation uniquement), on obtient une NLL moyenne de

$$\text{NLL}_{\text{GP-VAE}}(\text{cont}) = 0.4522 \quad \Longrightarrow \quad \text{PPL}_{\text{GP-VAE}}(\text{cont}) = 1.57,$$

ce qui confirme que, du point de vue interne du modèle, la continuation est très probable.

Pour évaluer la plausibilité linguistique des mêmes continuations du point de vue d'un modèle de langue externe, nous utilisons un GPT-2 small pré-entraîné comme juge. Pour un même prompt et une même longueur de continuation, nous comparons la variante latente-AR et l'ablation non-AR (bruit blanc latent). Les métriques suivantes sont calculées sous GPT-2 : NLL moyenne, perplexité, fraction de tokens rares (probabilité $< 10^{-4}$), taux de bigrammes répétés, taux de trigrammes répétés, taux de répétitions consécutives et fraction de caractères non-ASCII.

Ces chiffres mettent en évidence deux phénomènes. Premièrement, la variante non-AR est moins plausible du point de vue de GPT-2 : sa NLL moyenne est plus élevée de ≈ 2.17 nats par token, soit une perplexité externe environ 8.7 fois plus grande (3.8×10^5 contre 4.4×10^4). La fraction de tokens rares (probabilité $< 10^{-4}$ sous GPT-2) passe de 0.75 en autorégressivité dans l'espace latent à 0.94 en non-AR, ce qui signifie que l'ablation génère

Modèle	$\mathrm{NLL}_{\text{GPT-2}}$	$\mathrm{PPL}_{\text{GPT-2}}$	rare_frac	rep2	rep3	consec	non_ascii
GP-VAE AR	10.6847	4.37×10^4	0.750	0.032	0.000	0.048	0.000
GP-VAE non-AR	12.8507	3.81×10^5	0.938	0.000	0.000	0.000	0.008

TABLE 6 – Évaluation externe par GPT-2 d'une continuation de 64 tokens sur WIKITEXT-2, pour un même prompt (« The meaning of life ») et deux variantes de dynamique latente (GP-VAE AR vs GP-VAE non-AR). Les métriques sont calculées sous GPT-2 pré-entraîné.

beaucoup plus souvent des tokens que GPT-2 considère comme improbables. Deuxièmement, les métriques de répétition restent faibles dans les deux cas (répétition de bigrammes rep2 ≈ 0.03 pour AR et 0 pour non-AR, aucune trigramme répété, très peu de répétitions consécutives), ce qui indique que l'écart de qualité perçu par GPT-2 ne provient pas d'un mode de looping trivial, mais bien d'une meilleure adéquation globale de la structure latente AR à la distribution de texte.

En résumé, la dynamique latente autorégressive se traduit par des continuations que GPT-2 juge significativement plus plausibles que celles obtenues en supprimant toute corrélation temporelle dans le latent. Combiné à la perplexité interne très faible du GP-VAE AR sur cette continuation, ce résultat fournit une première validation quantitative du fait que le processus gaussien causal est effectivement exploité par le modèle, et qu'il joue un rôle déterminant dans la qualité linguistique des séquences générées.

6 Extension à WIKITEXT-103 et comparaison à un Transformer

Afin d'évaluer la robustesse des observations précédentes à l'échelle d'un corpus plus volumineux, nous avons étendu nos expériences à une version parquetisée de WIKITEXT-103. Le corpus brut comprend environ 1,809,468 lignes de texte issues de l'ensemble `train` de WIKITEXT-103. Le flux est concaténé puis découpé en blocs de longueur fixe, avec la même procédure de tokenisation GPT-2 que précédemment. Sauf mention contraire, les modèles partagent le même tokenizer GPT-2, la même longueur de contexte d'entraînement ($T_{\text{train}} = 64$ tokens) et le même budget d'optimisation (5 000 itérations).

Au cours de premières explorations sur WIKITEXT-103, nous avons constaté que certaines configurations naïves — en particulier un Transformer entraîné sur des blocs courts et avec un budget restreint — conduisaient à des

résultats difficiles à interpréter : perplexités de validation très élevées pour le Transformer et, inversement, compatibilité GPT-2 étonnamment faible pour les continuations du GP-VAE, reflétant davantage les biais structurels des juges autorégressifs que la qualité linguistique intrinsèque. Ces constats nous ont conduit à simplifier et harmoniser le protocole de manière à garantir une comparaison plus homogène entre architectures.

Dans cette section, nous présentons un protocole révisé sur WikiText-103 fondé sur deux modèles : (i) un GP-VAE-TCN+ à autorégressivité dans l'espace latent (AR), et (ii) un Transformer autorégressif sur les tokens, tous deux entraînés dans des conditions strictement symétriques. Nous introduisons en particulier une perplexité de *continuation intrinsèque* (teacher forcing sur des complétions tenues), utilisée comme métrique commune plus robuste que la seule perplexité de validation globale.

Rappel. Sauf mention contraire, les expériences sur WikiText-103 utilisent l'implémentation *TCN+*, caractérisée par un encodeur TCN dilaté non pyramidal et une implémentation vectorisée du prior GP.

6.1 Protocole expérimental harmonisé sur WikiText-103

Le corpus est obtenu en concaténant les fichiers `train` de WikiText-103, puis en les parquetisant et en les découpant en blocs de longueur $T_{\text{train}} = 64$ tokens après tokenisation GPT-2. Nous considérons deux modèles :

- **GP-VAE-TCN+ (latent AR).** Encodeur TCN dilaté étendu (non pyramidal), latent séquentiel $z_{1:T}$ régi par un processus gaussien causal, décodeur non-autorégressif parallèle. L'objectif d'apprentissage est une ELBO moyenne par token, avec divergence KL plafonnée et coefficient β adapté dynamiquement comme dans les expériences WikiText-2.
- **Transformer autorégressif sur les tokens.** Modèle de langage autorégressif opérant directement dans l'espace des tokens, entraîné avec le même budget d'itérations et la même longueur de contexte d'entraînement $T_{\text{train}} = 64$. Il a vocation à être un point de comparaison contrôlé pour analyser le comportement séquentiel et les phénomènes de collapse sur WikiText-103.

Les deux modèles partagent exactement le même flux de données (même tokenisation, mêmes blocs, mêmes splits train/validation) et le même budget d'optimisation (5 000 pas de gradient). L'objectif est d'isoler l'impact de l'architecture (latent corrélé vs autorégressif sur les tokens) plutôt que celui des différences de données ou de prétraitement.

6.2 Résultats internes : perplexité de validation et perplexité de continuation

Remarque méthodologique sur la comparabilité des perplexités. Il est essentiel de souligner que les perplexités reportées pour le GP-VAE et pour le Transformer ne reposent pas sur la même factorisation probabiliste et ne sont donc pas directement comparables. Dans le cas du GP-VAE, la perplexité de validation PPL(val) correspond à une *perplexité conditionnelle*, évaluée à partir de la distribution

$$p_\theta(x_{1:T} \mid z_{1:T}),$$

et dépend explicitement de la trajectoire latente globale $z_{1:T}$. À l'inverse, la perplexité de validation du Transformer repose sur une factorisation strictement autorégressive sur les tokens,

$$p(x_{1:T}) = \prod_{t=1}^{T} p(x_t \mid x_{<t}),$$

et mesure une quantité de nature différente.

Par conséquent, toute comparaison directe entre PPL(val) du GP-VAE et PPL(val) du Transformer serait méthodologiquement incorrecte. Dans cette étude, la comparaison entre architectures hétérogènes est donc effectuée exclusivement au moyen de la *perplexité de continuation intrinsèque* PPL(cont), évaluée en *teacher forcing* dans la factorisation propre à chaque modèle, ainsi que par l'analyse qualitative de la stabilité séquentielle et des phénomènes de collapse ou de répétition.

Sur WikiText-103, dans ce protocole harmonisé, le GP-VAE-TCN+ à autorégressivité dans l'espace latent atteint en validation les métriques suivantes :

$$\begin{aligned} \text{ELBO/token} &= -9.9237, \\ \text{LL}_0 &= -1.580, \\ \text{LL}_{\text{multi}} &= -6.696, \\ \text{KL}_{\text{eff}} &= 12.0 \text{ nats}, \\ \text{PPL(val)} &= 3.26. \end{aligned}$$

Sur le même flux et avec le même budget, le Transformer autorégressif sur les tokens obtient une perte de validation

$$\text{NLL(val)} = 6.4847 \quad \Longrightarrow \quad \text{PPL(val)} = \exp(6.4847) \approx 655.0.$$

Ces résultats indiquent que, dans le régime expérimental considéré, le GP-VAE-TCN+ autorégressivité dans l'espace latent obtient une perplexité de validation interne plus faible que celle du Transformer de référence. Cette observation doit être interprétée dans le cadre du protocole considéré, le Transformer servant ici de référence comportementale et non de point de comparaison état de l'art sur WikiText-103. Dans la suite de l'article, ce Transformer est donc explicitement positionné comme une *baseline*, utilisée exclusivement comme point de repère comportemental pour analyser des phénomènes de collapse, de répétition et de stabilité séquentielle, et non comme un compétiteur autorégressif optimisé.

Pour disposer d'une métrique plus homogène entre architectures, nous introduisons une perplexité de *continuation intrinsèque*, notée PPL(cont). Concrètement, à partir de prompts extraits du corpus, nous générons des complétions tenues de longueur fixe, puis ré-évaluons ces complétions en *teacher forcing* sous chaque modèle dans sa propre factorisation. Sur WikiText-103, cette métrique donne :

$$\begin{aligned} \mathrm{PPL(cont)}_{\text{GP-AR}} &= 3.17, \\ \mathrm{PPL(cont)}_{\text{GP-noAR}} &= 2.35, \\ \mathrm{PPL(cont)}_{\text{TF}} &= 14.12. \end{aligned}$$

Ce dernier chiffre ($\approx$ 14.1) place le Transformer dans la fourchette de perplexité typiquement reportée pour de petits modèles GPT-style sur WikiText-103 (ordre de grandeur 15–40). Autrement dit, même si sa perplexité de validation globale reste très élevée, la perplexité de continuation intrinsèque ne présente pas de comportement pathologique. Dans la suite de l'article, c'est donc principalement cette métrique de continuation que nous utilisons pour comparer les architectures, l'objectif étant d'analyser le rôle de l'autorégressivité dans l'espace latent à capacité fixe plutôt que de battre un Transformer fortement optimisé.

6.3 Évaluation externe par GPT-2 et biais structurels

Pour compléter l'analyse, nous utilisons un GPT-2 small pré-entraîné comme *juge externe* de plausibilité linguistique. Sur les mêmes prompts et complétions, nous calculons la perplexité et la fraction de tokens rares (probabilité $< 10^{-4}$) sous GPT-2. Pour les trois modèles considérés, nous

obtenons :

$$\begin{aligned} \mathrm{PPL}^{\text{GPT-2}}_{\text{GP-AR}} &= 162\,608.33, \\ \mathrm{PPL}^{\text{GPT-2}}_{\text{GP-noAR}} &= 284\,634.15, \\ \mathrm{PPL}^{\text{GPT-2}}_{\text{TF}} &= 108.96. \end{aligned}$$

avec des fractions de tokens rares

$$\begin{aligned} \text{rare\_frac}_{\text{GP-AR}} &= 0.859, \\ \text{rare\_frac}_{\text{GP-noAR}} &= 0.938, \\ \text{rare\_frac}_{\text{TF}} &= 0.109. \end{aligned}$$

Les métriques de répétition présentées ici (rep2, consec, etc.) sont calculées sur des continuations de longueur modérée ($L = 64$) dans le protocole harmonisé sur WikiText-103. Elles ne doivent pas être confondues avec les métriques de génération longue (jusqu'à $L = 3072$) utilisées pour l'expérience WikiText-2 de la Section 8.

Du point de vue de GPT-2, les séquences produites par le Transformer sont donc beaucoup plus probables que celles des deux variantes de GP-VAE, et contiennent moins de tokens rares. Cela illustre un biais structurel important : un juge externe autorégressif de type GPT-2 favorise naturellement les modèles de même famille (factorisation token-par-token) et pénalise les modèles latents non-autorégressifs, même lorsque ceux-ci obtiennent des perplexités internes très faibles sur leurs propres continuations.

Pour éclairer ces résultats, nous rapportons également des métriques de surface sur les complétions considérées, telles que le taux de bigrammes répétés rep2 et le taux de répétitions consécutives consec :

$$\text{rep2}_{\text{GP-AR}} = 0.000, \quad \text{rep2}_{\text{GP-noAR}} = 0.000, \quad \text{rep2}_{\text{TF}} = 0.254,$$

$$\text{consec}_{\text{GP-AR}} = 0.095, \quad \text{consec}_{\text{GP-noAR}} = 0.000, \quad \text{consec}_{\text{TF}} = 0.063.$$

Ces valeurs montrent qu'une perplexité GPT-2 faible ne coïncide pas nécessairement avec une diversité de surface maximale : le Transformer peut obtenir une perplexité externe très favorable tout en présentant des motifs de répétition plus marqués que les variantes latentes. Inversement, le GP-VAE AR, bien que fortement pénalisé en perplexité GPT-2, produit des séquences où les répétitions restent contrôlées.

Nous revenons plus en détail, en Section 10, sur les biais inhérents à l'usage de GPT-2 comme juge externe et sur la manière dont ces métriques doivent être interprétées dans le cadre d'une comparaison entre architectures hétérogènes.

6.4 Portée des résultats et positionnement de la contribution

Les expériences sur WikiText-103 menées dans ce protocole harmonisé permettent de préciser la portée de nos conclusions.

Premièrement, l'ensemble des résultats doit être lu dans ce régime : corpus de taille modérée (WikiText-2, WikiText-103), longueur de contexte d'entraînement fixe $T_{\text{train}} = 64$, budget d'entraînement limité (5 000 itérations) et modèles de taille restreinte.

Deuxièmement, la contribution principale de ce travail se situe dans l'*ablation contrôlée* du rôle de l'autorégressivité dans l'espace latent à capacité fixe, et non dans la recherche d'une performance compétitive face aux Transformers modernes. Le Transformer utilisé ici est explicitement positionné comme une baseline, destinée à fournir un point de repère pour la perplexité de continuation (par exemple $\text{PPL(cont)}_{\text{TF}} \approx 14.12$ sur WikiText-103) et pour illustrer des phénomènes de collapse et de looping. Des Transformers plus larges, mieux réglés et entraînés sur des corpus plus riches surpasseraient très probablement nos GP-VAE en perplexité brute.

Enfin, les résultats internes et externes doivent être articulés avec prudence : les métriques internes (ELBO, PPL(val), PPL(cont)) quantifient le comportement de chaque modèle dans sa propre factorisation, tandis que les métriques GPT-2 mesurent avant tout une *proximité statistique avec la famille des modèles autorégressifs*. Dans ce cadre, les expériences sur WikiText-103 fournissent un environnement contrôlé où l'on peut observer, de manière robuste, que la dynamique latente corrélée (autorégressivité dans l'espace latent) se comporte très différemment de son ablation non-AR et d'un Transformer — en termes de compatibilité avec le prior GP, de perplexité de continuation et de stabilité en génération longue.

7 Résultats sur WikiText-103 : GP-VAE-TCN+ vs Transformer

Implémentation utilisée. Sauf mention contraire, les expériences menées sur WikiText-103 utilisent l'implémentation *TCN+*, mieux adaptée aux corpus de grande taille et aux batchs volumineux en raison de son implémentation vectorisée du prior GP. Lorsque des résultats issus de l'implémentation pyramidale sont rapportés, ils sont explicitement signalés comme tels et utilisés à des fins de réplication ou de contrôle.

Cette section propose une vue synthétique des résultats internes obtenus sur WikiText-103 dans le protocole harmonisé décrit en Section 6. Nous y

résumons, pour le GP-VAE-TCN+ à autorégressivité dans l'espace latent et pour le Transformer autorégressif sur les tokens, les métriques de validation et de *continuation intrinsèque* (teacher forcing sur des complétions tenues).

Rappelons que les deux modèles sont entraînés sur une version parquetisée de WikiText-103 (fichiers `train/validation/test` concaténés puis découpés en blocs de $T_{\text{train}} = 64$ tokens), avec le même tokenizer GPT-2, la même longueur de contexte et le même budget d'optimisation (5 000 itérations). La comparaison porte donc sur l'architecture (latent corrélé vs autorégressif sur les tokens) plutôt que sur des différences de données ou de prétraitement.

Le Tableau 7 récapitule les principales métriques internes sur le jeu de validation et sur la tâche de continuation intrinsèque. Pour le GP-VAE-TCN+ AR, nous reportons l'ELBO moyenne par token, la perplexité de validation PPL(val) et la perplexité de continuation PPL(cont). Pour le Transformer, nous reportons la perte de validation NLL(val) (non affichée dans la colonne « ELBO/tok ») et les perplexités correspondantes.

Modèle	Paramètres	d_{mod}	d_z	ELBO/tok	PPL(val)	PPL(cont)
GP-VAE-TCN+ (AR latent)	18.3M	256	48	−9.92	3.26	3.17
Transformer autorégressif sur les tokens	30.5M	256	–	–	655.0	14.12

Table 7 – Résultats internes sur WikiText-103 avec blocs de $T_{\text{train}} = 64$ tokens, dans le protocole harmonisé de la Section 6. Le GP-VAE-TCN+ AR utilise un prior GP causal et un décodeur non-autorégressif parallèle ; le Transformer baseline est un modèle autorégressif opérant sur les tokens. Les deux modèles partagent le même tokenizer GPT-2, la même longueur de contexte et le même budget d'itérations.

Malgré une capacité paramétrique inférieure (18,3M de paramètres contre 30,5M pour le Transformer), le GP-VAE-TCN+ autorégressivité dans l'espace latent obtient une perplexité de validation PPL(val) $\approx$ 3,26 et une perplexité de continuation PPL(cont) $\approx$ 3,17, alors que le Transformer baseline reste à PPL(val) $\approx$ 655,0 et PPL(cont) $\approx$ 14,12 dans exactement les mêmes conditions de données et de budget. Autrement dit, dans ce régime, le modèle séquentiel latent fournit une vraisemblance interne meilleure qu'un Transformer autorégressif sur les tokens, tout en conservant un décodeur entièrement parallèle.

Ces résultats doivent toutefois être interprétés avec prudence. D'une part, le Transformer utilisé ici ne prétend pas être compétitif vis-à-vis des Transformers modernes sur WikiText-103 ; son rôle est celui d'une *baseline* servant de point de repère pour la perplexité de continuation et pour l'analyse des phénomènes de collapse. D'autre part, comme discuté en Section 10, les

métriques externes sous GPT-2 mesurent avant tout la compatibilité avec la famille des modèles autorégressifs et ne doivent pas être lues comme une mesure absolue de qualité linguistique. Dans ce cadre contrôlé, la Section 6 montre que la dynamique latente corrélée (autorégressivité dans l'espace latent) se distingue clairement de son ablation non-AR et du Transformer, en particulier sur la compatibilité avec le prior GP et sur la stabilité en génération longue.

7.1 Comparaison AR vs non-AR sur WikiText-103 ($L_{\mathbf{gen}} = 256$)

L'introduction d'une dynamique autorégressive dans le latent modifie profondément le comportement du modèle. Le GP-VAE à autorégressivité dans l'espace latent atteint une perplexité interne de PPL ≈ 3.19 (3.24 en continuation) et ne présente aucune boucle dégénérée, bien que ses séquences demeurent hors du support statistique effectif du juge GPT–2.

Sur les métriques internes, qui mesurent la cohérence probabiliste propre au modèle, on observe une supériorité de la variante AR :

$$\begin{aligned} \text{AR} &: \text{NLL} = 0.5022 \ (\text{PPL} = 1.65), \\ \text{non-AR} &: \text{NLL} = 0.7335 \ (\text{PPL} = 2.08). \end{aligned}$$

Le latent autorégressif apprend des trajectoires significativement plus cohérentes et mieux alignées avec le prior GP causal, tandis que la variante non-AR adopte une structure plus bruitée et moins prédictive.

Les métriques structurelles corroborent cette analyse :

$$\begin{aligned} \text{rep2}_{\text{AR}} = 0.016, \quad \text{rep3}_{\text{AR}} = 0.000, \quad \text{consec}_{\text{AR}} = 0.063, \\ \text{rep2}_{\text{noAR}} = 0.000, \quad \text{rep3}_{\text{noAR}} = 0.000, \quad \text{consec}_{\text{noAR}} = 0.032. \end{aligned}$$

La variante non-AR produit un bruit plus uniforme, tandis que l'AR introduit une dynamique temporelle légère mais stable, sans collapse ni boucles.

Enfin, le juge GPT–2 attribue aux deux variantes des perplexités très élevées (de l'ordre de 10^5), reflet non pas de leur qualité, mais du biais structurel d'un évaluateur autorégressif sur les tokens appliqué à un modèle parallèle à processus gaussien dans l'espace latent. Cette observation confirme l'inadéquation des métriques AR classiques pour évaluer un modèle à autorégressivité purement latente.

7.2 Comparaison AR vs non-AR sur WikiText-103 ($L_{\text{gen}} = 512$)

Sur WikiText-103 avec un contexte étendu à $L\_gen = 512$, le GP-VAE à autorégressivité dans l'espace latent apprend des régularités internes stables et maintient une perplexité intrinsèque faible (PPL ≈ 3.3). Lorsqu'on évalue la continuation dans la propre distribution du modèle, la variante AR surpasse la version non-AR : le GP-AR atteint une NLL de 0.4793 (PPL = 1.61), contre 0.7355 (PPL = 2.09) pour le GP-noAR. Le latent autorégressif adopte une dynamique temporelle cohérente et mieux alignée avec le prior GP causal, tandis que la variante non-AR reste plus bruitée. Les métriques de répétition de surface confirment cette tendance : le GP-AR présente une structure séquentielle légère mais stable (rep2 = 0.032, rep3 = 0.000, consec = 0.095), tandis que le GP-noAR se rapproche d'un bruit i.i.d. (rep2 = 0.000, consec = 0.000). Aucun cas de boucle catastrophique n'a été observé, même à $L\_gen = 512$, ce qui valide directement l'hypothèse centrale du modèle.

Du point de vue d'un juge autorégressif sur les tokens comme GPT–2, les deux variantes de GP-VAE apparaissent pourtant improbables : elles se voient attribuer des perplexités de l'ordre de 10^5, tandis qu'un Transformer atteint une perplexité GPT–2 de 133 malgré une PPL interne médiocre (≈ 395). L'écart GP-AR vs. GP-noAR est non monotone : le GP-AR, avec une `rare_frac` plus élevée (≈ 0.875), produit des séquences plus atypiques et structurées mais étrangères au support de GPT–2, alors que le GP-noAR génère un bruit plus uniforme, rejeté pour des raisons différentes. Cette dissociation entre perplexités internes et évaluation AR sur les tokens montre que les métriques classiques des modèles autorégressifs ne peuvent servir de critère pertinent pour juger un modèle parallèle à processus gaussien dans l'espace latent. Le GP-VAE–AR met en place une dynamique séquentielle stable et prédictive, tandis que la variante non-AR se limite à un bruit faiblement structuré, bien que les deux demeurent en dehors des régions de forte masse de probabilité de la distribution autorégressive apprise par GPT–2.

7.3 Comparaison AR vs non-AR sur WikiText-103 ($L_{\text{gen}} = 1024$)

À $L\_gen = 1024$, les deux variantes du GP-VAE restent stables, mais leurs comportements divergent de manière systématique. La version latente autorégressive conserve une perplexité interne légèrement plus faible (PPL

≈ 3.33 contre 3.37 pour la variante non-AR), indiquant une meilleure compatibilité avec le prior gaussien causal. Cette différence, réduite par rapport aux horizons plus courts, reflète le fait que l'autorégressivité impose une dynamique latente cohérente même lorsque la variance cumulée du GP augmente avec la longueur de la séquence.

Les métriques structurelles confirment cette tendance : la variante AR présente une légère irrégularité temporelle (consec ≈ 0.095) sans répétitions dégénérées, tandis que la version non-AR reste essentiellement stationnaire (rep2 = rep3 = 0), caractéristique d'un latent i.i.d. lissé. Dans les deux cas, aucune boucle catastrophique n'est observée, ce qui suggère que la dynamique imposée par le GP conserve un régime de stabilité jusqu'à 1024 pas de temps.

Enfin, le contraste avec un Transformer est marqué : malgré une perplexité interne très élevée, ce dernier est favorisé par le juge GPT-2, confirmant que les métriques AR sur les tokens ne sont pas adaptées pour évaluer un modèle parallèle à processus gaussien dans l'espace latent. L'ensemble de ces observations renforce l'idée que l'autorégressivité dans l'espace latent joue un rôle déterminant dans l'alignement structurel avec le prior et dans la robustesse séquentielle à long horizon.

7.4 Comparaison AR vs non-AR sur WikiText-103 ($L_{\text{gen}} = 2048$)

À $L\_gen = 2048$, le GP–VAE demeure stable malgré un horizon séquentiel très étendu. La validation donne un ELBO/tok de -10.21, une perplexité interne de 3.68 et une perplexité de continuation de 3.81, sans aucune instabilité de la KL ni collapse. Cette stabilité à long contexte est notable pour un modèle variationnel séquentiel.

Métriques internes. Les évaluations *own-model continuation* mettent en évidence un écart clair entre les deux dynamiques latentes :

$$\begin{aligned} \text{GP-AR}: &\ \text{NLL} = 0.4621\ (\text{PPL} = 1.59), \\ \text{GP-noAR}: &\ \text{NLL} = 0.7599\ (\text{PPL} = 2.14), \\ \text{TF}: &\ \text{NLL} = 2.9170\ (\text{PPL} = 18.49). \end{aligned}$$

Comme aux horizons plus courts, le latent autorégressif fournit des trajectoires plus compatibles avec le prior gaussien causal. Lorsque la longueur latente augmente, la corrélation imposée par le prior pénalise davantage la variante non-AR, dont le comportement proche du bruit i.i.d. devient statistiquement incompatible. L'écart AR vs non-AR, subtil à $L\_gen = 1024$, réapparaît ici de manière significative.

Structure séquentielle. Les métriques superficielles confirment cette tendance :

$$\begin{aligned} \text{GP-AR} &: \text{rep2} = 0.016, \quad \text{rep3} = 0, \quad \text{consec} = 0.111, \\ \text{GP-noAR} &: \text{rep2} = 0, \quad \text{rep3} = 0, \quad \text{consec} = 0. \end{aligned}$$

Le latent AR manifeste une structuration faible mais réelle, sans boucle dégénérée, tandis que la variante non-AR reste quasi stationnaire. Le Transformer, en contraste, présente des répétitions AR classiques.

Métriques AR sur les tokens. Le juge GPT–2 attribue aux deux GP–VAE des perplexités très élevées ($\sim 10^5$), alors qu'il favorise fortement le Transformer. Ces valeurs reflètent la nature non-autorégressive du GP–VAE et confirment que les métriques AR classiques sont inadaptées pour évaluer un modèle parallèle à processus gaussien dans l'espace latent, en particulier à long horizon.

Synthèse. Cette expérience montre que l'autorégressivité dans l'espace latent reste robuste et exploite de mieux en mieux le prior gaussien lorsque la longueur du chemin latent augmente. À $L\_gen = 2048$, le GP–VAE AR conserve une cohérence interne, tandis que la variante non-AR devient plus fortement discordante avec la structure imposée par le prior. Le Transformer, pour sa part, se dégrade, bien qu'il reste favorisé par les métriques AR externes.

7.5 Comparaison AR vs non-AR sur WikiText-103 ($L_{\text{gen}} = 3072$)

Sur WikiText-103 avec un horizon très long ($L\_gen = 3072$), le GP-VAE à autorégressivité dans l'espace latent reste stable : on observe une perplexité interne de validation PPL ≈ 3.83 et une perplexité de continuation PPL ≈ 3.96, sans divergence de la KL (maintenue à 12 nats) ni collapse séquentiel.

Les métriques intrinsèques de continuation confirment l'avantage de l'autorégressivité dans l'espace latent sur la variante non-AR, tandis que le Transformer baseline reste moins cohérent dans sa propre distribution :

La variante latente autorégressive maintient donc des trajectoires significativement plus prévisibles que le latent i.i.d., même à très long horizon, tandis que le Transformer se dégrade fortement en termes de structure probabiliste interne. Comme aux autres échelles de contexte, le juge GPT-2 attribue des

Modèle	NLL	PPL
GP-VAE AR	0.6840	1.98
GP-VAE non-AR	0.7809	2.18
Transformer	3.6684	39.19

TABLE 8 – Métriques de continuation propre au modèle sur WIKITEXT-103 ($L_{\text{gen}} = 3072$).

perplexités massives aux deux GP-VAE (PPL de l'ordre de 10^5) tout en favorisant le Transformer (PPL ≈ 125), ce qui confirme que des métriques autorégressives sur les tokens ne constituent pas un critère pertinent pour évaluer un modèle parallèle à processus gaussien dans l'espace latent.

8 Comparaison directe sur WIKITEXT-2 : GP-VAE AR vs Transformer baseline

Nous complétons l'analyse précédente par une expérience contrôlée sur WIKITEXT-2, dans laquelle nous confrontons directement le GP-VAE à autorégressivité dans l'espace latent à un Transformer autorégressif sur les tokens.

Le protocole est le suivant. Le corpus WIKITEXT-2 RAW est chargé via l'API `datasets`, tokenisé avec le tokenizer GPT-2, puis concaténé et recoupé en blocs de longueur fixe. Le GP-VAE à autorégressivité dans l'espace latent utilise exactement le même encodeur TCN, le même décodeur parallèle et le même schéma d'apprentissage (ELBO, cap KL à 8 nats, annealing de β jusqu'à $\approx 0{,}35$) que dans les expériences précédentes.

Perplexité : distinctions méthodologiques essentielles. Les perplexités reportées pour le GP-VAE sont des perplexités conditionnelles (voir Section 4.3) et ne reflètent pas une factorisation autorégressive. À l'inverse, les perplexités du Transformer mesurent bien la quantité $p(x_t \mid x_{<t})$. Ces deux métriques ne sont donc **pas directement comparables**. La comparaison pertinente se fait sur :

— les métriques de continuation (NLL, PPL, cohérence),
— le comportement qualitatif (répétitions, rare_frac),
— et l'effet de l'autorégressivité dans l'espace latent.

Sur ce corpus, le GP-VAE à autorégressivité dans l'espace latent atteint

en fin d'entraînement une perplexité interne de validation de

$$\mathrm{PPL(val)} \approx 2.25, \quad \mathrm{ELBO/token} \approx -8.27,$$

avec une perplexité d'entraînement stabilisée autour de 3–4 et une KL/token saturée au cap fixé. Cette valeur doit être interprétée comme un indicateur *interne* de cohérence séquentielle du latent, et non comme une mesure comparable à celle d'un modèle autorégressif.

Le Transformer baseline, quant à lui, obtient une perte de validation d'environ 3.18 nats/token, soit une perplexité autorégressive d'environ

$$\mathrm{PPL(val)} \approx 24.0.$$

Ce score constitue une mesure standard de performance dans le cadre autorégressif, mais ne doit pas être mis en regard de la perplexité interne du GP-VAE. Malgré une baisse rapide de la perte d'entraînement (perplexité d'entraînement ≈ 20), le modèle reste limité en généralisation.

Au-delà de cette métrique globale, nous évaluons les deux modèles avec des métriques de génération détaillées, en utilisant un GPT-2 pré-entraîné comme juge externe. À partir de prompts extraits de l'ensemble de test, nous générons des continuations de longueur $L \in \{32, 64, 128, 256, 512, 1024, 2048, 3072\}$ et mesurons :

- des métriques de répétition dans le texte généré (taux de bigrammes/trigrammes répétés, fraction de tokens appartenant à une boucle exacte, fraction de séquences catastrophiques) ;
- la perplexité des continuations sous GPT-2, la fraction de tokens rares (probabilité $< 10^{-4}$) et l'auto-similarité moyenne/maximale de fenêtres d'embeddings GPT-2.

Dans cette expérience, les métriques de répétition sont calculées sur des générations libres de longueur $L \in \{32, 64, 128, 256, 512, 1024, 2048, 3072\}$, à partir de prompts extraits de l'ensemble de test de WikiText-2. Les phénomènes de looping rapportés pour le Transformer (par exemple rep2 ≈ 1 et loop_frac ≈ 1 dès $L = 32$) se réfèrent à ce protocole de *génération longue*, distinct du protocole harmonisé de la Section 6.

Les résultats mettent en évidence un contraste très net. Du côté du GP-VAE à autorégressivité dans l'espace latent, les taux de répétition restent modérés pour toutes les longueurs considérées : par exemple, pour des continuations de longueur $L = 32$ à $L = 1024$, le taux de bigrammes répétés rep2 reste typiquement compris entre 0,06 et 0,30, les trigrammes répétés rep3 entre 0,01 et 0,15, et la fraction de tokens appartenant à une boucle exacte reste quasi nulle (loop_frac ≈ 0). Aucune séquence catastrophique

n'est observée jusqu'à $L = 3072$. Du point de vue de GPT-2, ces continuations restent en revanche très surprenantes : les perplexités externes sont élevées (de l'ordre de 10^3 à 10^5 selon L) et la fraction de tokens rares importante (souvent entre 0,3 et 0,8). L'auto-similarité moyenne des fenêtres d'embeddings reste strictement inférieure à 1, ce qui confirme une diversité réelle du contenu généré.

Le Transformer baseline adopte un comportement inverse. Dès $L = 32$, les métriques de répétition saturent : les taux de bigrammes et trigrammes répétés sont très proches de 1, la fraction de tokens appartenant à une boucle exacte vaut loop_frac ≈ 1, et toutes les séquences sont marquées comme catastrophiques (cat_frac ≈ 1). Le modèle tombe donc immédiatement dans un mode de looping déterministe, où un motif court est répété indéfiniment. Paradoxalement, ce comportement pathologique est *récompensé* par le juge GPT-2 : les perplexités externes sont faibles (souvent proches de 1–2), la fraction de tokens rares est quasi nulle, et l'auto-similarité des fenêtres d'embeddings est exactement 1 (les fenêtres successives sont identiques). Autrement dit, le Transformer minimise efficacement la perplexité en apprenant une trajectoire très probable sous GPT-2, mais au prix d'une diversité linguistique presque totalement annihilée.

Cette expérience confirme deux points essentiels. Premièrement, le GP-VAE à autorégressivité dans l'espace latent maintient des séquences longues variées et dépourvues de boucles catastrophiques, contrairement au Transformer baseline, qui tombe rapidement dans un mode dégénéré de répétition. Deuxièmement, les métriques classiques de type perplexité ne suffisent pas à caractériser la qualité générationnelle : un modèle peut obtenir une perplexité très faible tout en produisant des sorties dégénérées, comme l'illustre clairement le Transformer. Le GP-VAE à autorégressivité dans l'espace latent, en imposant une dynamique corrélée dans le latent et en s'appuyant sur un décodeur parallèle, offre un compromis plus robuste entre cohérence statistique interne et diversité séquentielle.

9 Analyse centrale : AR vs non-AR dans le latent

Nous résumons ici les trois diagnostics internes qui mettent en évidence le rôle crucial de la dynamique autorégressive dans le latent.

(A) Compatibilité avec le prior GP. La log-densité moyenne des trajectoires latentes évaluées sous le prior gaussien corrélé est plus élevée pour la variante autorégressivité dans l'espace latent ($\log p_{\mathrm{GP}}(z) \approx +2.3 \times 10^3$),

indiquant une bonne compatibilité avec le processus gaussien causal. À l'inverse, l'ablation non-AR est associée à une log-densité faible ($\log p_{\mathrm{GP}}(z) \approx -2.7 \times 10^7$), traduisant une incompatibilité avec le prior corrélé.

L'écart de log-densité entre les deux régimes atteint ainsi environ 2.7×10^7 nats, confirmant que les trajectoires hors régime autorégressivité dans l'espace latent sont incompatibles avec la géométrie induite par le prior GP.

(B) Structure temporelle des latents. L'auto-corrélation cosinus moyenne en fonction du lag montre que la variante AR présente une forte corrélation à court terme (lag 1 : $\rho \approx 0{,}57$), décroissant progressivement avec le lag, signature d'un processus latent lisse. La variante non-AR reste proche de zéro pour tous les lags ($\rho \approx 0{,}00$), ce qui correspond à du bruit i.i.d.

(C) Amplitude des variations latentes. La norme moyenne des pas $\|z_t - z_{t-1}\|$ révèle que la variante AR produit des trajectoires stables ($4{,}87 \pm 0{,}98$), avec des variations modérées d'un pas au suivant, tandis que la variante non-AR effectue des sauts plus brusques ($8{,}46 \pm 0{,}74$), caractéristiques d'un bruit non structuré.

Synthèse. Ces trois diagnostics convergent : l'autorégressivité dans l'espace latent ne se contente pas d'améliorer marginalement la compatibilité avec le prior, elle transforme qualitativement le comportement du modèle en imposant une structure temporelle cohérente, absente dans le régime non-AR.

10 Remarque méthodologique : biais structurel lié à l'usage de GPT-2 comme juge externe

Il est important de souligner que l'évaluation de la plausibilité linguistique repose sur un modèle externe, GPT-2, dont la factorisation est strictement autorégressive. Ce choix est méthodologiquement cohérent — il permet de comparer sur une base commune des modèles dont les factorisations internes diffèrent — mais il introduit un biais structurel qu'il convient d'expliciter.

En effet, GPT-2 estime la probabilité d'un texte selon une distribution $p_\theta(x_t \mid x_{<t})$ apprise au cours de son pré-entraînement. Les séquences qui respectent les régularités locales propres aux Transformers autorégressifs (structure token-par-token, dépendances locales prévisibles, rythmes statistiques caractéristiques des modèles AR) seront systématiquement jugées plus probables. À l'inverse, les séquences issues de modèles non-autorégressifs, comme le GP-VAE étudié ici, ne suivent pas cette factorisation et ne sont

pas contraintes par les mêmes biais inductifs : leur dynamique est globale, portée par un latent corrélé, et non par un mécanisme de prédiction locale.

Par conséquent, l'usage d'un juge de type GPT-2 introduit un *biais de compatibilité* : un modèle autorégressif sera évalué favorablement non seulement parce qu'il produit du texte plausible, mais aussi parce que sa structure interne coïncide avec celle du modèle évaluateur. Inversement, un modèle génératif non-autorégressif peut obtenir une perplexité interne très faible tout en demeurant pénalisé par GPT-2, non pas en raison d'une incohérence linguistique absolue, mais parce que sa factorisation ne correspond pas à celle du juge.

Ce biais ne remet pas en cause la validité de l'évaluation — GPT-2 fournit une métrique stable et reproductible — mais il en limite la portée : la perplexité GPT-2 ne doit pas être interprétée comme une mesure absolue de qualité linguistique, mais comme un indicateur de *proximité statistique avec la famille des modèles autorégressifs*. C'est pourquoi, dans cette étude, les comparaisons GPT-2 sont utilisées pour mesurer la compatibilité relative des variantes (AR vs non-AR) et non pour établir une hiérarchie globale entre architectures hétérogènes.

11 Synthèse globale des résultats

Les expériences menées sur WikiText-2 et WikiText-103 mettent en évidence un contraste net entre les dynamiques latentes autorégressives et non-autorégressives au sein d'un même GP-VAE, à architecture et capacité fixées.

Lorsque le latent est régi par un processus gaussien causal, le modèle apprend des trajectoires temporellement corrélées, compatibles avec le prior, et conserve une stabilité séquentielle sur de très longues générations. À l'inverse, l'ablation non-AR conduit à des latents proches d'un bruit indépendant, incompatibles avec la structure du GP, et à une dégradation systématique du comportement à long horizon.

Ces observations sont reproduites de manière cohérente sur deux implémentations indépendantes du modèle et sur deux corpus de tailles distinctes. Elles suggèrent que, dans le régime étudié, la dynamique latente joue un rôle central dans la cohérence séquentielle, indépendamment de la complexité du décodeur.

12 Discussion

Il est important de souligner que les conclusions de ce travail ne reposent pas sur une implémentation particulière du GP-VAE, mais sur des phénomènes observés de manière cohérente à travers deux implémentations indépendantes, utilisées selon les contraintes propres à chaque corpus et protocole expérimental.

Un point central mis en lumière par cette étude est que l'autorégressivité dans l'espace latent n'agit pas comme une simple régularisation additionnelle, mais comme un mécanisme structurant qui conditionne qualitativement le comportement séquentiel du modèle. Lorsque cette dynamique est supprimée, le GP-VAE conserve une capacité de reconstruction locale mais perd toute cohérence temporelle globale, révélant une dissociation entre performance locale et stabilité à long horizon.

Ce résultat est conceptuellement intéressant car il montre qu'une partie substantielle de la séquentialité peut être déplacée hors de l'espace symbolique des tokens et encodée dans une dynamique continue latente, sans recourir à une factorisation autorégressive explicite dans le décodeur. Dans ce cadre, le prior gaussien causal ne se contente pas de contraindre la norme des latents : il impose une géométrie temporelle qui guide l'apprentissage et stabilise la génération.

L'opposition observée entre cohérence interne et évaluations externes autorégressives met également en évidence une limite méthodologique des métriques classiques. Un modèle peut être pénalisé par un juge autorégressif non pas en raison d'une incohérence linguistique intrinsèque, mais parce que sa factorisation diffère fondamentalement de celle du modèle évaluateur. Cette dissociation souligne la nécessité de critères d'évaluation adaptés aux modèles génératifs parallèles à dynamique latente.

Enfin, ces résultats suggèrent que la complexité séquentielle d'un modèle de langage ne réside pas uniquement dans la profondeur ou la capacité du décodeur, mais aussi dans la structure probabiliste imposée à l'espace latent. Cette perspective ouvre la voie à des architectures hybrides où la dynamique est portée par le latent, tandis que le décodeur peut rester léger et parallèle.

13 Conclusion et perspectives

Ce travail constitue une étude de faisabilité empirique du paradigme d'autorégressivité purement latente dans les modèles de langage. À architecture et capacité contrôlées, nos expériences montrent que la dynamique imposée dans

l'espace latent conditionne de manière décisive le comportement séquentiel du modèle.

Lorsque le latent suit un processus gaussien causal, le GP-VAE apprend des trajectoires temporellement corrélées, compatibles avec le prior, et conserve une stabilité remarquable en génération longue. À l'inverse, l'ablation non-autorégressive, bien que soumise au même objectif variationnel, conduit à des latents proches d'un bruit indépendant et à une dégradation systématique de la cohérence à long horizon.

Ces différences latentes se traduisent directement dans l'espace du texte généré. Le régime autorégressif latent maintient une diversité effective et évite les boucles catastrophiques, tandis que la variante non-AR s'effondre dès que l'horizon augmente. Le contraste observé avec un Transformer autorégressif sur les tokens souligne par ailleurs les limites des métriques classiques de perplexité, qui peuvent favoriser des modèles localement prédictibles mais globalement dégénérés.

Dans le cadre expérimental considéré, ces résultats suggèrent qu'une dynamique latente corrélée peut porter une part significative de la séquentialité, indépendamment d'un décodeur autorégressif lourd. L'autorégressivité dans l'espace latent ne joue pas ici un rôle accessoire, mais constitue le mécanisme central assurant la compatibilité avec le prior, la structure des trajectoires et la stabilité des générations longues.

Perspectives. L'extension de ce travail à des dynamiques latentes plus expressives, à des contextes plus longs et à des architectures hybrides combinant latent corrélé et autorégressivité symbolique légère constitue une direction naturelle pour de futurs travaux. Une évaluation plus large, incluant des métriques humaines ou orientées tâche, permettra également de dépasser les biais inhérents aux juges autorégressifs.

Références

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[2] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Józefowicz, and S. Bengio. Generating sentences from a continuous space. In *Conference on Computational Natural Language Learning (CoNLL)*, 2016.

[3] M. Fraccaro, S. K. Sønderby, U. Paquet, and O. Winther. Sequential neural models with stochastic layers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[4] C. K. Sønderby, T. Raiko, L. Maae, S. K. Sønderby, and O. Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[5] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[6] V. Fortuin, G. Rätsch, and S. Klakow. GP-VAE : Deep probabilistic time series imputation. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

[7] J. Bayer and C. Osendorfer. Learning Stochastic Recurrent Networks. In *International Conference on Learning Representations (ICLR), Workshop Track*, 2015.

[8] J. Chung, K. Kastner, L. Dinh, K. Goel, A. Bengio, and H. Courville. A Recurrent Latent Variable Model for Sequential Data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[9] J. Gu, J. Bradbury, C. Xiong, V. O. K. Li, and R. Socher. Non-Autoregressive Neural Machine Translation. In *International Conference on Learning Representations (ICLR)*, 2018.

[10] M. Ghazvininejad, O. Levy, Y. Liu, and L. Zettlemoyer. Mask-Predict : Parallel Decoding of Conditional Masked Language Models. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

[11] Y. Ruffenach. Modèle de langage GP-VAE à autorégressivité dans l'espace latent. In *Zenodo Preprint*, 2025. `https://zenodo.org/records/17696132`.