ChatGPT

Exercice guidé : comprendre les paramètres T-P-L-S-F-C-V sur un même prompt

Introduction

Les modèles génératifs comme ChatGPT n'ont pas qu'une **consigne** ; ils disposent également de paramètres qui influencent fortement la sortie. Pour vous entraîner, nous allons utiliser **un seul exemple de texte et un seul prompt**, puis modifier un paramètre à la fois. Vous verrez comment ces réglages transforment le résultat, du plus factuel au plus créatif. Les paramètres à connaître sont :

- **T** (*Temperature*) : règle le niveau de créativité. Une température basse donne des réponses prévisibles et concentrées, tandis qu'une température élevée favorise la diversité et l'originalité [1] .
- **P** (*Top-p*) : limite l'ensemble des mots possibles lors de chaque choix. Un **top-p** élevé permet plus de diversité, alors qu'un **top-p** bas concentre la sortie sur les mots les plus probables [2] .
- **L** (*max tokens*) : limite le nombre maximum de mots/tokens générés [3] .
- **S** (*Stop sequence*) : définit une ou plusieurs séquences qui arrêtent la génération lorsque le modèle les écrit [4] .
- **F** (*Format*) : impose la structure de la réponse (Markdown, liste à puces, JSON…).
- **C** (*Contexte*) : précise les sources autorisées pour répondre afin d'éviter les hallucinations.
- **V** (*Validation*) : ce sont vos règles de vérification : longueur maximale, nombre de points, absence de hors sujet, mention des incertitudes…

Exemple de base

Extrait à résumer

> **Extrait :** « Les panneaux solaires convertissent la lumière en électricité grâce à l'effet photovoltaïque. Leur coût a fortement baissé depuis 2010, ce qui a accéléré leur adoption. Les limites majeures sont l'intermittence de la production et le besoin de stockage ou d'appoint. »

Prompt de référence

> **Prompt :** « Résume l'extrait ci-dessus en **5 puces** (≤ **120 mots**), et **cite 2 limites** de l'étude. Utilise **uniquement** cet extrait. Si une information manque, écris *"Incertitude : …"*. Termine par "### FIN". »

Nous allons exécuter ce même prompt en modifiant un paramètre à la fois.

Étape 1 : Paramètres de base

Commencez par fixer un ensemble de paramètres **standards** :

- **Temperature (T)** : 0,2 (factuel, peu de créativité).
- **Top-p (P)** : 0,85 (équilibre entre précision et variété).
- **max_tokens (L)** : 200 (suffisant pour 5 puces + 2 limites).
- **Stop sequence (S)** : `["### FIN"]` (le modèle s'arrête dès qu'il écrit cette séquence).
- **Format (F)** : liste en Markdown (5 puces, puis une ligne "Limites : …").
- **Contexte (C)** : "Réponds uniquement à partir de l'extrait fourni."
- **Validation (V)** : vérifier que la réponse contient 5 puces, $\leq$120 mots, 2 limites, aucune information hors extrait, et "Incertitude" si nécessaire.

Faites tourner le prompt avec ces paramètres. Vous devriez obtenir un résumé concis et factuel des phrases principales.

Étape 2 : Température (T)

Gardez tous les paramètres de base et modifiez **uniquement** la température :

1. **T = 0,1** (très basse) : la sortie est presque mécanique. Elle suit exactement le format et répète les termes de l'extrait. Cela assure la précision, mais le style peut sembler robotique.
2. **T = 0,5** (intermédiaire) : la sortie reste correcte, mais elle peut reformuler légèrement ("convertissent la lumière" → "captent l'énergie solaire") et ajouter des connecteurs. C'est adapté si vous souhaitez une lecture plus naturelle.
3. **T = 0,9** (élevée) : le modèle prend des risques et propose des tournures plus imagées (par exemple "caprices du soleil" pour l'intermittence). Cela peut enrichir le texte, mais attention aux dérives hors contexte [1].

À faire : exécutez trois fois le prompt en changeant uniquement **T** (0,1 ; 0,5 ; 0,9). Comparez les différences : observe-t-on davantage d'images ou de synonyme ? Le message reste-t-il fidèle à l'extrait ? Notez vos conclusions.

Étape 3 : Top-p (P)

Revenez à **T = 0,2**, puis variez le **top-p** :

1. **P = 0,5** : le modèle ne choisit que parmi les mots les plus probables. La sortie est concise, parfois répétitive.
2. **P = 0,85** (valeur de base) : l'équilibre entre précision et diversité.
3. **P = 1,0** : tous les mots restent envisageables. La réponse peut intégrer des expressions plus variées et moins attendues [2].

À faire : testez ces trois valeurs de **top-p** en gardant **T = 0,2** et les autres paramètres identiques. Notez comment la variété du vocabulaire change.

Étape 4 : Longueur maximale (L)

Changez la valeur de **max_tokens** pour limiter ou étendre la longueur :

1. **L = 80** : le modèle doit condenser davantage ; certaines informations risquent de disparaître ou d'être résumées.
2. **L = 200** (valeur de base) : confortable pour 5 puces.
3. **L = 350** : la réponse peut devenir plus développée, avec des reformulations ou des détails supplémentaires.

Rappel : `max_tokens` fixe la longueur maximale de la sortie [3]. Testez ces trois valeurs et observez comment la densité d'information varie.

Étape 5 : Stop sequence (S)

Retirez ou modifiez la séquence d'arrêt. Cela permet de voir comment le modèle termine naturellement ou s'il déborde :

1. **Sans stop sequence** : la génération continue jusqu'à atteindre `max_tokens`. Le modèle peut dépasser le format demandé (par exemple, il pourrait ajouter des phrases après les puces).
2. **Stop = ["### FIN"]** (valeur de base) : la génération s'arrête net quand le modèle écrit "### FIN" [4].

Vous pouvez essayer d'autres séquences ("FIN", "**FIN**") pour vérifier que le modèle obéit correctement à l'instruction.

Étape 6 : Format (F)

Modifiez la structure de sortie :

1. **Markdown à puces** (notre format de base) : 5 tirés + ligne "Limites : ...".
2. **Liste numérotée** : remplacez les puces par des numéros (1., 2., 3., ...). Le contenu reste identique, mais la présentation change. Idéal pour des étapes.
3. **JSON structuré** : imposez un schéma, par exemple :

```
{
  "resume": [ "puce 1", "puce 2", "puce 3", "puce 4", "puce 5" ],
  "limites": [ "limite 1", "limite 2" ]
}
```

Le modèle doit alors fournir un objet JSON valide ; l'utilisation d'une stop sequence `"}"` peut aider à couper proprement après l'accolade fermante.

Exercice : modifiez le paramètre `response_format` ou la consigne de format pour tester ces trois structures.

Étape 7 : Contexte (C)

Changez l'étendue du contexte autorisé :

1. **Contexte strict** (base) : "Réponds uniquement à partir de l'extrait fourni." Le modèle doit se limiter aux informations présentes ; il doit écrire "Incertitude : …" si quelque chose manque.
2. **Contexte libre** : supprimez cette phrase. Le modèle peut alors puiser dans ses connaissances internes et compléter le texte (par exemple, citer le taux de baisse du prix des panneaux solaires ou l'année exacte de la découverte de l'effet photovoltaïque). Cela peut enrichir la réponse mais introduire des risques d'erreur ou de hors sujet.

À faire : testez les deux versions et comparez. Les sorties diffèrent-elles par le degré de détail ? Y a-t-il des informations que l'extrait ne contient pas ?

Étape 8 : Validation (V)

Définissez une liste de vérifications à appliquer à chaque sortie :

- **Longueur totale** $\leq$ 120 mots (vous pouvez compter approximativement).
- **Exactement 5 puces et 2 limites**.
- **Aucune information hors extrait** (ou toute info manquante est signalée par "Incertitude : …").
- **Structure conforme** au format choisi (Markdown ou JSON).

Après chaque génération, vérifiez la sortie manuellement ou via un script. Ajustez les paramètres si l'une de ces règles n'est pas respectée.

Récapitulatif des effets

Le tableau ci-dessous résume l'effet attendu de chaque paramètre ; il ne contient que des mots-clés pour rester lisible :

Paramètre	Effet clé	Quand augmenter ?	Quand diminuer ?
T (Temperature)	Créativité : plus haute → diversité, images [1]	Pour du storytelling, brainstorming	Pour un devoir factuel ou codé
P (Top-p)	Diversité du vocabulaire [2]	Si le style est répétitif ou monotone	Si vous voulez une sortie très concentrée
L (max_tokens)	Longueur maximale [3]	Pour autoriser plus de détails	Pour forcer une réponse brève
S (Stop)	Coupe nette à une séquence [4]	Pour encadrer un format (JSON, liste)	Inutile si la longueur suffit
F (Format)	Structure (Markdown, JSON…)	Lorsqu'un format est imposé ou lisible	—

Paramètre	Effet clé	Quand augmenter ?	Quand diminuer ?
C (Contexte)	Sources autorisées	Pour éviter les hallucinations	À proscrire si vous voulez que l'IA complète
V (Validation)	Contrôle de qualité	Toujours : définir des règles claires	—

Conclusion

En gardant **le même prompt** et **le même extrait**, modifier un paramètre à la fois permet de comprendre l'impact de chaque réglage. Une température élevée et un top-p large favorisent la créativité mais augmentent le risque de hors sujet, tandis qu'une température basse avec un top-p réduit produit des réponses plus stables et factuelles [2]. `max_tokens` fixe la longueur totale de la sortie [3], et la stop sequence garantit un arrêt propre lorsque la séquence apparaît [4].

Pour vos exercices, commencez avec une configuration "factuelle" (T = 0,2 ; P = 0,85 ; L = 200 ; stop = ["### FIN"]) et modifiez un paramètre à la fois. Observez, comparez et notez comment la sortie change. C'est en expérimentant que l'on maîtrise ces leviers et qu'on apprend à choisir la bonne combinaison pour chaque situation.

---

[1] [2] [3] Complete Guide to Prompt Engineering with Temperature and Top-p
https://promptengineering.org/prompt-engineering-with-temperature-and-top-p/

[4] Top 7 LLM Parameters to Instantly Boost Performance
https://www.analyticsvidhya.com/blog/2024/10/llm-parameters/