

Documentation HostoMytho

Bertrand REMY

28 décembre 2024

Résumé

Documentation technique détaillant les logiques d'HostoMytho et les données générées.

1 Objectifs de l'application

HostoMytho est un "jeu ayant un but", développé dans le cadre du projet ANR. Il a pour but de permettre à des utilisateurs d'annoter des compte-rendus médicaux synthétiques (générés automatiquement par différents modèles de langage). Les joueurs qualifient la plausibilité des textes (en évaluant la qualité de la langue et le réalisme médical), identifient et classifient les erreurs (médicales, linguistiques, ou autres), et collectent d'autres données langagières. Les annotations produites par les joueurs sont utilisées pour affiner les modèles de langage grâce à des techniques de fine-tuning.

2 Logique de fonctionnement des jeux

Les jeux affichent des textes ou des passages de textes, dont la longueur est paramétrable. Plusieurs cas de figure sont possibles :

- **Textes générés** : La majorité des textes sont générés, et les annotations à faire dessus ne sont pas connues à l'avance. Les réponses sont donc toujours affichées comme bonnes aux joueurs
- **Textes de contrôle** : Certains textes de contrôle, avec des réponses connues, sont affichés. Ils permettent d'évaluer les joueurs. En cas de mauvaise réponse, le joueur entre dans une boucle de tests et doit fournir une réponse correcte. Cela permet de détecter discrètement les comportements suspects. Les mauvaises réponses font baisser la note de fiabilité des joueurs, les bonnes réponses la font augmenter.
- **Textes déjà traités** : Des textes déjà annotés par d'autres joueurs sont affichés pour renforcer les annotations existantes, et comparer les réponses des joueurs entre eux. Ces textes impactent également la fiabilité des joueurs, mais moins fortement.

2.1 Poids des réponses

Les réponses des utilisateurs ont un poids, déterminé par leur taux de fiabilité (indice sur 100). Ce poids renforce la pertinence des annotations : par exemple, un joueur avec une fiabilité de 90 attribue un poids de 90 à ses réponses, permettant ainsi de calculer des moyennes pondérées. De plus, un joueur médecin (statut déclaré lors de la création du compte) voit ses réponses avoir un supplément de 30% sur le poids correspondant à sa fiabilité.

Exemple de plausibilité moyenne d'un texte, calculée avec les réponses des joueurs dans MythoOuPas :

- Joueur 1 (**J1**) : Fiabilité = 100, Réponse donnée = 75%.
- Joueur 2 (**J2**) : Fiabilité = 50, Réponse donnée = 25%.
- Joueur 3 (**J3**) : Fiabilité = 40, Réponse donnée = 0%.

1. Somme pondérée :

$$\text{Somme pondérée} = \sum (\text{plausibilité} \times \text{poids})$$

$$J1 : 75 \times 100 = 7500 \quad J2 : 25 \times 50 = 1250 \quad J3 : 0 \times 40 = 0$$

$$\text{Total} : 7500 + 1250 + 0 = 8750$$

2. Moyenne pondérée :

$$\text{Moyenne de fiabilité pondérée} = \frac{\text{Somme pondérée}}{\text{Somme des poids}} = \frac{8750}{100 + 50 + 40} = 46.05$$

La plausibilité moyenne du texte est donc de **46.05%**.

2.2 Calcul des gains de points

Chaque jeu a une base de gain de points paramétrable, par exemple 5 pour MythoNo. Cette base peut-être augmentée si le joueur fait plusieurs sélections sur un texte dans MythoNo ou MythoOu-Pas. Chaque sélection ajoute 2 points de base. Cette base de points est ensuite multipliée par l'indice de fiabilité du joueur / 80, puis multiplié par le coefficient multiplicateur du joueur. Ce coefficient correspond à $1 + 0.1$ par hauts faits du joueur.

Exemple :

Un joueur ayant une fiabilité de 90 et 6 hauts faits validés réalise deux annotations sur un texte dans MythoNo :

$$\begin{aligned} \text{Points gagnés} &= (\text{Base de points} + 2 \times \text{Annotations}) \times \frac{\text{Fiabilité}}{80} \times (1 + 0.1 \times \text{Hauts faits}) \\ &= (5 + 2 \times 2) \times \frac{90}{80} \times 1.6 = 16 \end{aligned}$$

Le joueur gagne **16 points**.

2.3 Augmentation du pourcentage de chance d'attraper un criminel

Si le joueur a un taux de fiabilité inférieur à 30, il ne gagne plus de pourcentage (pour éviter les spams et les tricheurs). Plus le joueur attrape de criminels, plus le gain diminue. Le calcul se fait par rapport à un exponentiel et à un taux de décroissance arbitraire de 0.14.

Formule :

$$\text{Gain (\%)} = \text{Base de points} \times e^{-0.14 \times \text{Criminels attrapés}}$$

Exemple : Un joueur ayant déjà attrapé 3 criminels, avec une base de 5 :

$$\text{Gain} = 5 \times e^{-0.14 \times 3} = 5 \times 0.657 = 3.285$$

Le joueur gagne **3.285** points de pourcentage supplémentaires.

3 Explication des données générées

Les textes sont sauvegardés dans la table `texts`. Toutes les phrases de ces textes sont stockées dans la table `sentences`, qui contiennent le `text_id` du texte auquel elles appartiennent, ainsi que la position de ces phrases dans le texte.

Chaque mot (ou token) est sauvegardé dans la table `tokens`, où figurent le `text_id` et le `sentence_id` pour identifier leur origine. Cette table contient également le champ `position`, permettant de localiser le mot dans la phrase à laquelle il appartient.

Ce fonctionnement permet de travailler sur des ensembles de phrases, et non uniquement sur des textes entiers, ces derniers pouvant être longs et fastidieux à traiter pour les joueurs. Les annotations de plausibilité effectuées dans *MythoOuPas* permettent ainsi d’isoler les passages problématiques d’un texte. Cette table contient toutes les annotations de typage des joueurs dans MythoTypo, avec une clé étrangère vers la table précédente, et une vers les différents types possibles.

Les données spécifiques à chaque jeu sont stockées dans des tables distinctes. Voici un visuel et une description de ces tables :

user_sentence_specification - Annotations de négations/absences (MythoNo)

Cette table contient les annotations de négation et d’absence effectuées par les joueurs dans *MythoNo*. Elle fait référence au texte affiché via le champ `text_id`, et inclut la liste des positions des mots sélectionnés dans le texte (`word_positions`). Elle contient également le poids de l’annotation, calculé en fonction du statut et de la fiabilité du joueur.

user_text_rating - Notations de plausibilité (MythoOuPas)

Cette table contient toutes les notations de plausibilité des joueurs dans *MythoOuPas*, exprimées sous forme de pourcentages : 0, 25, 50, 75 ou 100. Elle inclut également le poids de chaque réponse, ainsi que la liste des positions des phrases affichées (`sentence_positions`). De plus, la table contient une clé étrangère (`group_id`) faisant référence à la table `group_text_rating`.

group_text_rating - Groupement des notations (MythoOuPas)

Cette table regroupe plusieurs notes d’utilisateurs concernant un même ensemble de phrases d’un même texte. Elle calcule la moyenne pondérée des pourcentages de plausibilité donnés par les joueurs, en tenant compte du poids de chaque réponse. Le champ `average_plausibility` contient cette moyenne, et le champ `votes_count` comptabilise le nombre total de votes ayant contribué à cette moyenne. Quand on veut charger un texte déjà joué dans *MythoOuPas* pour comparer les réponses des utilisateurs, on récupère un texte dans cette table.

user_error_details - Erreurs détectées (MythoOuPas et MythoTypo)

Cette table est alimentée progressivement par les données collectées dans *MythoOuPas*. Elle répertorie les erreurs sélectionnées par les joueurs dans ce jeu. Ces données sont ensuite réutilisées dans *MythoTypo*, où les erreurs identifiées dans *MythoOuPas* sont récupérées. Pour afficher une erreur, le texte correspondant est localisé grâce au champ `text_id`, et l’erreur est mise en surbrillance à l’aide du champ `word_positions`.

À savoir : les erreurs de contrôle à trouver dans le jeu sont également stockées dans cette table. Les champs `is_test`, `test_error_type_id` et `reason_for_rate` ne sont remplis que dans le cas d’une erreur de contrôle.

user_typing_errors - Annotations de typage (MythoTypo)

Cette table contient toutes les annotations de typage réalisées dans *MythoTypo*. Elle inclut :

- `user_error_details_id` : une clé étrangère vers la table `user_error_details`.
- `error_type_id` : une référence aux différents types d’erreurs possibles.

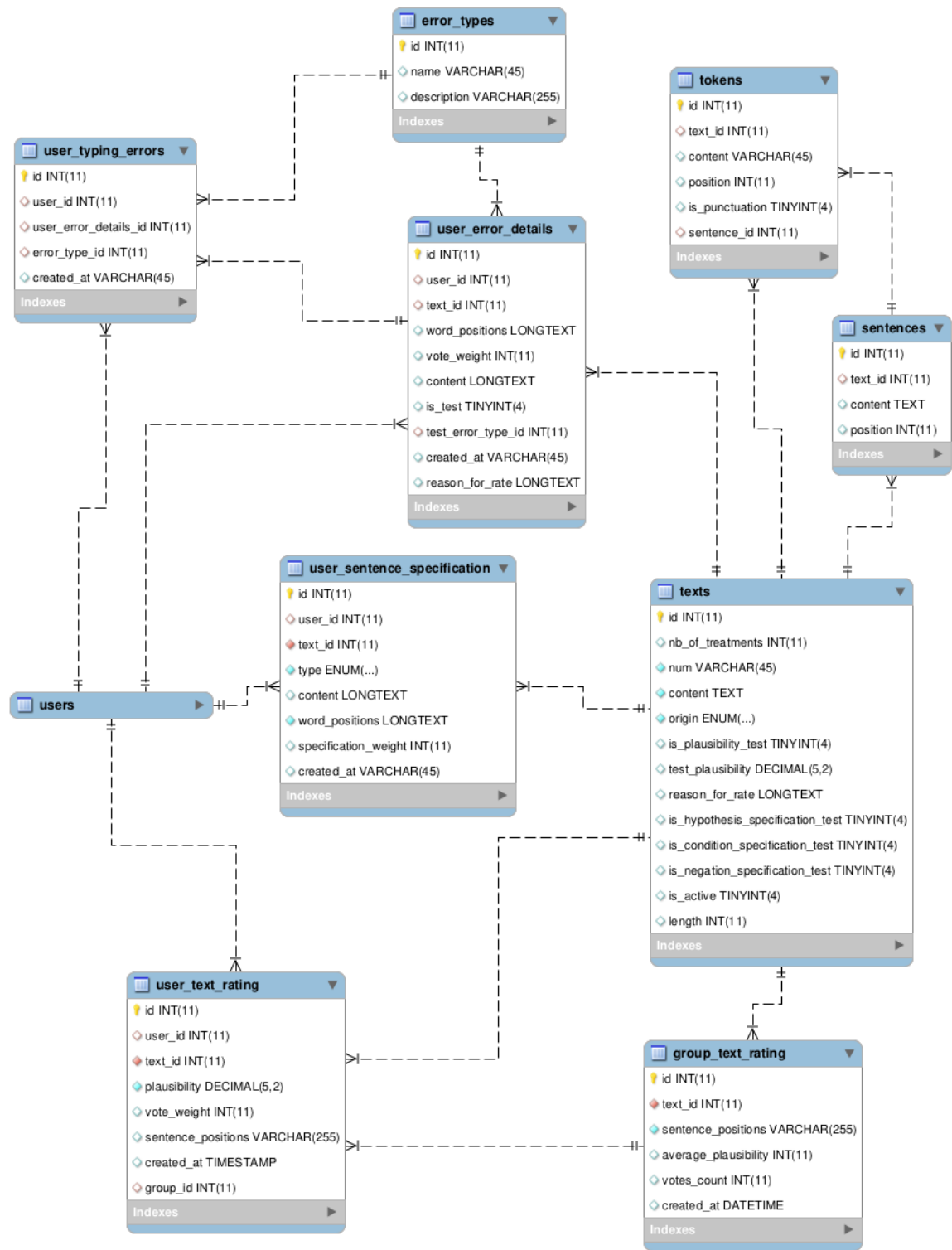


FIGURE 1 – Tables d’annotations