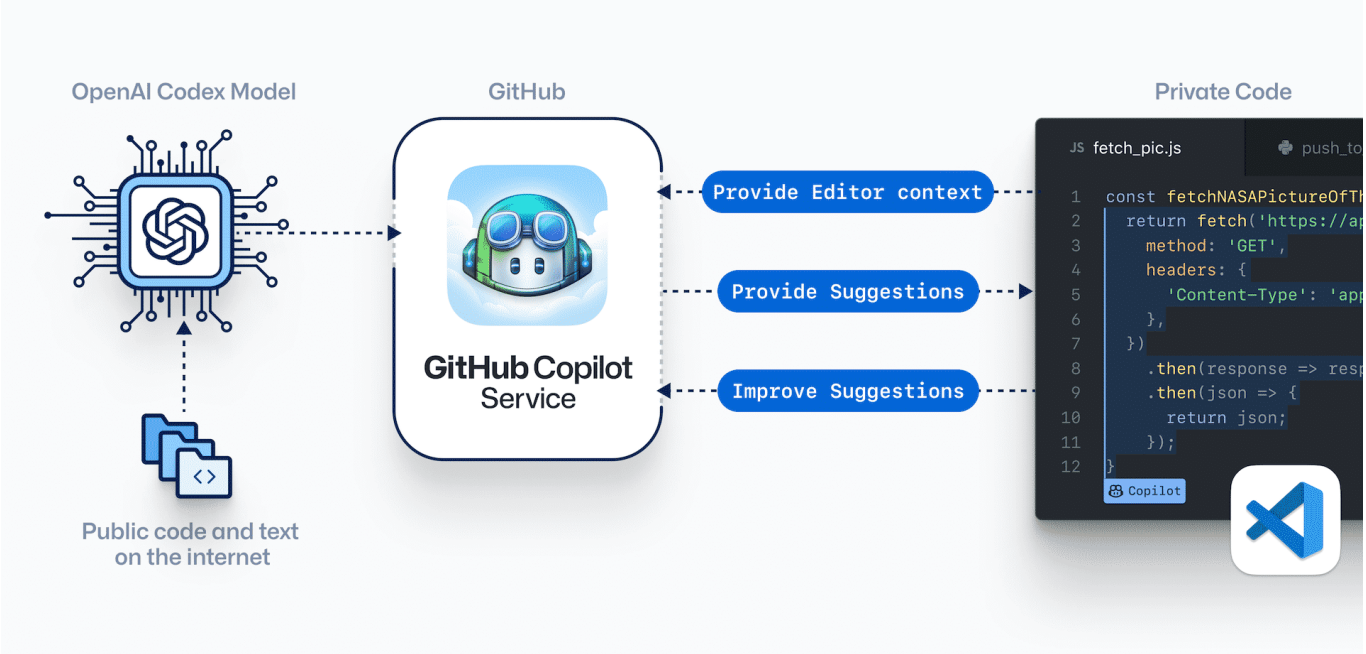
GitHub Copilot suscite de l'agitation chez les développeurs

« Ils ont fini par trouver un moyen de me faire écrire des commentaires. » Une remarque certes empreinte d'humour, mais symbolique de l'impression que GitHub Copilot laisse à ceux qui ont pu l'expérimenter.

L'outil se présente sous la forme d'une extension pour Visual Studio Code. Il met en œuvre le principe de la complétion de code. Mais en s'appuyant sur un modèle d'apprentissage automatique développé avec OpenAI. Ce modèle, nommé Codex, a été entraîné sur des dépôts publics de code source et sur du texte en anglais. En l'état, Python, JavaScript, TypeScript, Ruby et Go sont ses langages « de prédilection ».



GitHub expérimentait jusqu'alors en cercle fermé. Il ouvre désormais la démarche sur invitation, pour un « nombre limité » de testeurs. Tout en prenant soin de rappeler que le code généré requiert – *a minima* – la même attention que celui que produisent les développeurs.

> *We spent the last year working closely with OpenAI to build GitHub Copilot. We've been using it internally for months, and can't wait for you to try it out; it's like a piece of the future teleported back to 2021. https://t.co/NwZNIF8s7H*
>
> *— Nat Friedman (@natfriedman) June 29, 2021*

Copyright et données personnelles

Au-delà de l'incohérence potentielle du code qu'il suggère*, Copilot présente encore bien des limites. Par exemple, l'incapacité à aller chercher du contexte au-delà du fichier sur lequel on travaille. Et même à exploiter pleinement ce fichier s'il est trop long.

Autre écueil : la divulgation de données personnelles. GitHub l'estime très peu probable, mais ne l'exclut pas, puisqu'il en existe dans les dépôts utilisés pour l'entraînement. Une parade est en place pour le moment : un filtre censé empêcher l'affichage d'adresses e-mail au format standard.

Il reste également du travail pour éviter les termes choquants et les références à des composants obsolètes ou à des pratiques insécurisées. Mais aussi sur la question du copyright. Il arrive en l'occurrence à Copilot de reprendre des fragments de code tels quels. En particulier lorsqu'il manque de contexte ou qu'il existe une solution « universelle » à un problème. GitHub en est conscient et assure développer un système de signalement.

> *Another @Speakeasy_JS example.*
>
> *Here you can see that #GitHubCopilot figured out that the getCurrentEvent() function should either return the event, or null if the event is not valid. pic.twitter.com/JhBYfSuLTW*
>
> *— Feross (@feross) June 30, 2021*

Copilot deviendra, à terme, un produit commercial. Dans le portefeuille « aide aux développeurs » de Microsoft, il cohabite avec IntelliCode. Intégré à Visual Studio, cet outil recourt aussi au *machine learning*. Mais essentiellement en local, au travers du *runtime* ONNX.

> *Hi Warren. IntelliCode and Copilot teams have been collaborating closely, to provide a « better together » experience. However, the underlying tech isn't the same. Copilot's powered by OpenAI Codex, for rich code synthesis via a cloud service.IntelliCode uses multiple local models*
>
> *— Mark Wilson-Thomas (@MarkPavWT) June 29, 2021*

** GitHub fournit un indicateur de performance, fondé sur le masquage de fonctions Python courantes dans les dépôts d'entraînement. Copilot en a reconstitué 43 % du premier coup (et 57 % en ayant droit à 10 essais).*

Illustration principale © Florian Olivo – Unsplash