

Enhancing Your Visual Studio 2026 Copilot Experience: Tips and Workarounds

TechRounder PDF Edition

Live article: <https://www.techrounder.com/how-to/enhancing-your-visual-studio-2026-copilot-experience/>

By Vipin PG | Published January 22, 2026 | Updated March 10, 2026 | Format: Guide | 3 min read

Quick answer

To improve Copilot in Visual Studio 2026, first update Visual Studio, then confirm Copilot is enabled in your GitHub Copilot settings and in Visual Studio under Tools > Options > Extensions > Copilot (reinstall if it's missing). For better model support, use VS Code for AI-heavy tasks and check for extension conflicts.

Key points

The article argues that while Visual Studio 2026 Copilot is a promising productivity tool, its current implementation can be frustrating, especially because Visual Studio 2026 doesn't yet support the latest AI models like Gemini 3 that are available in VS Code, which can push developers to bounce between IDEs and break their workflow. It recommends starting with the basics by making sure Visual Studio 2026 is fully up to date, noting that Microsoft frequently ships updates that can improve compatibility and performance. The author specifically points readers to Visual Studio's built-in update path-opening Visual Studio 2026, going to the Help menu, and selecting "Check for Updates"-as a simple step that often resolves underlying issues. From there, it suggests ensuring Copilot is correctly enabled and configured (including in GitHub's Copilot settings) so you can get the most out of the features that are currently supported, even without Gemini 3.

If you've been struggling with the Visual Studio 2026 Copilot, you're not alone. I've spent quite a bit of time wrestling with it myself. While it's a promising tool, its current implementation leaves a lot to be desired. Let me share some practical steps and insights that can help you improve your Copilot experience in Visual Studio 2026.

Understanding the Current Limitations

Before diving into solutions, understand the limitations we're facing. The major issue is that Visual Studio 2026 doesn't yet support the latest AI models like Gemini 3, which are available in VS Code. This discrepancy often forces developers to switch between IDEs, disrupting workflow and productivity.

Setting Up Your Environment

First things first, ensure that your Visual Studio 2026 is up to date. Microsoft frequently releases updates that can improve compatibility and performance. Here's how you can check for updates:

1. Open Visual Studio 2026.
2. Go to Help in the menu bar.
3. Select Check for Updates.
4. Follow the prompts to install any available updates.

This might seem basic, but you'd be surprised how often updates can resolve underlying issues.

Enabling Copilot Features

Once you're updated, let's ensure Copilot is properly set up. While Gemini 3 isn't available in Visual Studio yet, you can still make the most of what's currently supported.

Enable Copilot in GitHub

1. Go to your GitHub account settings.
2. Navigate to Copilot settings.
3. Ensure that Copilot is enabled and check for any available AI model options.

Even though Gemini 3 isn't available, you might find other models that can improve your coding experience.

Configure Copilot in Visual Studio

Next, verify that Copilot is configured correctly in Visual Studio:

1. Open Visual Studio and go to Tools > Options.
2. Select Extensions > Copilot.
3. Ensure that the Copilot extension is enabled and running.

If you don't see Copilot options, you might need to reinstall the extension from the Visual Studio Marketplace.

Optimizing Your Workflow

While we're waiting for better integration, you can still optimize your workflow using these strategies:

Use VS Code for Specific Tasks

If you find that certain tasks are better supported in VS Code, don't hesitate to switch over for those specific tasks. For instance, tasks that heavily rely on AI model suggestions might perform better in VS Code.

Use Other AI Tools

Consider integrating other AI tools that are compatible with Visual Studio 2026. This can help fill the gap left by the lack of newer models. Tools like TabNine or Kite might offer some of the predictive capabilities you need.

Troubleshooting Common Issues

If you're still experiencing issues, here are some common troubleshooting steps:

Check for Extension Conflicts

Sometimes, other extensions can interfere with Copilot:

1. Go to Tools > Extensions and Updates.
2. Disable other extensions one by one and check if Copilot's performance improves.

Monitor Performance

Visual Studio can sometimes lag due to high CPU or memory usage. Keep an eye on your system's performance and consider upgrading your hardware if necessary.

Looking Ahead

Microsoft is aware of these issues, and improvements are likely on the horizon. Meanwhile, staying proactive with updates and configurations can make a notable difference. By optimizing your current setup and keeping an eye on emerging solutions, you can maintain productivity and reduce frustration.

Hang in there, and remember that today's challenges with Visual Studio 2026 are paving the way for a more smooth future.