

BCS KI Pugin

Norman Steinhoff

Version vom 24. Februar 2025

Inhaltsverzeichnis

1	Entwicklung	1
1.1	Gemini Nano	1
1.2	Dokumentation von Gemini Nano	2
1.3	Crome Canary	2
1.4	Notwendige Voraussetzungen für Gemini Nano	2
1.5	Chrome Extensions	3
1.6	TypeScript	3
1.7	Wie haben wir Typescript genutzt?	4

1 Entwicklung

Im Rahmen dieses Software-Projekts sollten wir eine Chrome-Extension entwickeln, die das Ticketsystem der Firma Projektron um KI-Funktionen erweitert. Dazu sollten wir einerseits ein lokales Modell (on-device) und ein externes Modell verwenden.

Wir haben Gemini Nano als lokales und Llama-3.3 als externes Modell verwendet.

1.1 Gemini Nano

Gemini Nano ist ein von Google entwickeltes Sprachmodell [1]. Der Hauptzweck von Gemini Nano ist es, KI Funktionen clientseitig ausführen zu können, ohne dafür jedesmal ein Sprachmodell herunterladen zu müssen. Dazu wurde Gemini Nano direkt in Chrome Canary integriert. [2]

Aktuell befinden sich Gemini Nano und die dazugehörige API in einer Testphase. In diesem Zustand können und haben sich ¹ deren Funktionen geändert. [2]

Die Gemini API ist in verschiedene Task API's aufgeteilt. Aktuell sind dies [3]:

- LanguageDetection
- Translation

¹Wie wir mehrfach während der Entwicklung unseres Plugins feststellen mussten

- Summary
- Writer + Rewriter
- PromptApi ²

1.2 Dokumentation von Gemini Nano

Um die oben genannten API's nutzen zu können sollte man dem Origin Trial beitreten. Dies ging über [4]. Hier erhielten wir die jeweils neuesten Informationen via Email. Dies war notwendig, da die eigentliche Dokumentation [5, 6, 7, 8, 9] seit längerer Zeit nicht mehr aktuell ist (Stand 04.02.2025). Inzwischen gibt es zwar über [2] eine neue öffentlich zugängliche Dokumentation, allerdings ist diese ebenfalls nicht vollständig.

1.3 Chrome Canary

Chrome Canary ist die experimentelle Version von Googles Chrome Browser und bietet die Möglichkeit, die neuesten Funktionen zu testen. Canary kann als eine Art Labor betrachtet werden, in dem neue Ideen und Features ausprobiert werden, bevor sie in die endgültige Version von Chrome aufgenommen werden. Diese Vorabversion wird täglich aktualisiert, wodurch die neuesten Entwicklungen stets verfolgt werden können. Da es sich jedoch um eine experimentelle Version handelt, kann es vorkommen, dass Canary instabil ist oder Fehler aufweist [10].

Zum Zeitpunkt der Entwicklung unserer Extension war Chrome Canary die einzige Möglichkeit Gemini Nano zu benutzen.

1.4 Notwendige Voraussetzungen für Gemini Nano

Gemini Nano ist unter Windows (10, 11) und unter MacOS (>=13) verfügbar, sofern man mindestens 22 GB freien Speicher hat, über eine GPU verfügt und >= 6 GB VideoRAM hat.

Um Gemini Nano in Chrome Canary nutzen zu können müssen dann mehrere Flags gesetzt werden. Dazu gibt man in der Adresszeile `chrome://flags` ein. Dort müssen dann folgende Flags gesetzt werden:

- Enables optimization guide on device -> *Enabled BypassPerfRequirement*
- Text Safety Classifier ³ -> *Disabled*
- Language detection web platform API -> *Enabled*
- Experimental translation API -> *Enabled without language pack limit*

²Unser Plugin stützt sich letztlich auf die PromptApi, auch wenn wir zwischendurch die anderen API's ausprobiert haben.

³Wenn der Text Safety Classifier nicht ausgeschaltet ist, weigert sich Gemini Nano Texte zu verarbeiten, die nicht in Englisch sind.

- Summarization API for Gemini Nano -> *Enabled*
- Writer API for Gemini Nano -> *Enabled*
- Rewriter API for Gemini Nano -> *Enabled*
- Prompt API for Gemini Nano -> *Enabled*

Desweiteren sollte man danach in der Adresszeile `chrome://components/` eingeben und dann *Optimization Guide On Device Model* suchen und aktualisieren.

Um zu testen, ob alles korrekt eingestellt ist und ob man die notwendigen Voraussetzungen erfüllt kann man nun in der Console des Chrome Browsers ⁴ folgendes eintippen: `await self.ai.languageModel.availability()` falls man nun `after-download` oder `available` als Antwort erhält, kann man Gemini Nano prinzipiell verwenden. Sollte man jedoch `no` als Antwort bekommen, so wird der verwendete Computer nie in der Lage sein Gemini Nano auszuführen.

1.5 Chrome Extensions

Chrome-Extensions [11] sind kleine Softwareprogramme, die die Funktionalität des Google Chrome-Browsers erweitern und an die individuellen Bedürfnisse des Nutzers anpassen. Sie stellen eine Bereicherung des Browser-Erlebnisses dar, indem sie zusätzliche Funktionen implementieren, bestehende modifizieren oder repetitive Aufgaben automatisieren.

Chrome-Extensions basieren auf Webtechnologien wie HTML, CSS und JavaScript. Somit sind sie prinzipiell in jedem Texteditor erstellbar.

1.6 TypeScript

Typescript (TS) ist eine von Microsoft entwickelte Erweiterung von JavaScript (JS). [12]

TS wird zu JS convertiert, wodurch der geschriebene Code in nahezu jedem Browser läuft. [12]

TS erweitert JS durch ein Typsystem in dem unter anderem Interfaces und Klassen beschrieben werden können. Dadurch reduzieren sich die möglichen Fehlerquellen enorm, da der TS-Compiler einfache Typ-Überprüfungen vollziehen kann. Desweiteren ist dadurch auch eine bessere Integration mit IDE's wie zum Beispiel VSCode gegeben. [12]

TS ist dabei so entworfen worden, dass jeder gültige JS-Code auch gültiger TS-Code ist. Es ist also möglich ein bestehendes JS-Projekt ohne weiteres auf TS umzustellen. Anschließend kann man, wenn man möchte, den vorhandenen Code zu TS convertieren. [12] [13] Dabei hilft ein Feature von TS. Die so genannte `tsconfig.json` Datei enthält Einstellungen für den TS-Compiler um beispielsweise erweitertes ErrorChecking zu ermöglichen. [14]

⁴auf einem Mac: `Darstellung -> Entwickler -> Java-Script-Konsole`

1.7 Wie haben wir Typescript genutzt?

Aufgrund der vielen Vorteile von TS haben wir unsere Chrome Extension direkt und vollständig (100 %) in TS geschrieben.

Außerdem haben wir die tsconfig.json Datei so eingestellt, dass der TS-Compiler maximal streng vorgeht. Dies bedeutet, dass wir so viele Warnungen und Vorschläge vom Compiler bekommen möchten, wie irgend möglich. Außerdem wurde eingestellt, dass jede Warnung als Fehler zu interpretieren ist und zum Abbruch führt. Für weitere Informationen zur tsconfig.json kann man auf der offiziellen Webseite [14] nachsehen .

Literatur

- [1] Google. Artificial intelligence | chrome for developers. <https://developer.chrome.com/docs/ai>. Accessed: 2025-02-04.
- [2] Google. Built-in ai | ai on chrome | chrome for developers. <https://developer.chrome.com/docs/ai/built-in>. Accessed: 2025-02-04.
- [3] Google. Built-in ai apis | ai on chrome | chrome for developers. <https://developer.chrome.com/docs/ai/built-in-apis>. Accessed: 2025-02-04.
- [4] Google. Join the prompt api for chrome extensions origin trial | blog | chrome for developers. <https://developer.chrome.com/blog/prompt-api-origin-trial>. Accessed: 2025-02-04.
- [5] Google. The language detection api - update 4 - google docs. <https://docs.google.com/document/d/1lY40hdaWizzImXaI2iCGto9s0Y6s25BcDJDYQvxpvk4/edit?tab=t.0>. Accessed: 2025-01-01.
- [6] Google. The translation api - update 6 - google docs. https://docs.google.com/document/d/1bzpeKk4k26KfjtR-_d90uXLMpJdRMiLZA0VNMuFiejk/edit?tab=t.0. Accessed: 2025-01-01.
- [7] Google. The summarization api - update 3 - google docs. <https://docs.google.com/document/d/1Bvd6cU9VIEb7kHTAOCtmmHNAYlIZdeNmV70y-2CtimA/edit?tab=t.0>. Accessed: 2025-01-01.
- [8] Google. The writer and rewriter apis - update 5 - google docs. https://docs.google.com/document/d/1WZ1AvfrIWDwzQXdqIcCOTcrWLGgmoesN1VGFbKU_D4/edit?tab=t.0. Accessed: 2025-01-01.
- [9] Google. Welcome and heads-up about the prompt api - update 1 - google docs. https://docs.google.com/document/d/1VG8HIyz361zGduWgNG7R_R8Xkv000J8b5C9QKeCjU0c/edit?pli=1&tab=t.0#heading=h.drihdh1gpv8p. Accessed: 2025-01-01.
- [10] Google. Chrome canary-funktionen für entwickler– google chrome. <https://www.google.com/chrome/canary/>. Accessed: 2025-02-04.

- [11] Google. Chrome extensions | chrome for developers. <https://developer.chrome.com/docs/extensions>. Accessed: 2025-01-01.
- [12] Microsoft. Typescript: Javascript with syntax for types. <https://www.typescriptlang.org>. Accessed: 2025-02-04.
- [13] Microsoft. Typescript: Documentation - typescript for the new programmer. <https://www.typescriptlang.org/docs/handbook/typescript-from-scratch.html>. Accessed: 2025-02-04.
- [14] Microsoft. Typescript: Tsconfig reference - docs on every tsconfig option. <https://www.typescriptlang.org/tsconfig/>. Accessed: 2025-02-04.