

# **POLONISTA v2.0**

---

**Asystent Prostego Języka dla LibreOffice Writer z integracją NVIDIA Bielik-11B**

## **Streszczenie**

**POLONISTA działa jak tłumacz**, który zamienia trudne teksty na proste. Tak jak Google Translate tłumaczy między językami, POLONISTA "tłumaczy" skomplikowany tekst urzędowy na język zrozumiały dla każdego.

**Co robi program:** Zaznaczasz tekst → Klikasz przycisk → Tekst staje się prostszy.

**Dla kogo:** Dla osób piszących dokumenty, które muszą być zrozumiałe dla wszystkich.

## **Spis Treści**

1. Wymagania systemowe
2. Pobieranie klucza API
3. Instalacja makra
4. Konfiguracja klucza API
5. Użytkowanie programu
6. Rozwiązywanie problemów
7. Proponowane zmiany dla localwriter
8. Dokumentacja techniczna
9. Bibliografia

## 1. Wymagania systemowe

### ✓ Co musisz mieć

Element	Wymaganie	Jak sprawdzić
System operacyjny	Windows 10/11	Menu Start → Ustawienia → System → Informacje
LibreOffice Writer	Wersja 7.0 lub nowsza	Pomoc → O programie LibreOffice
Internet	Stałe połączenie	Otwórz przeglądarkę i wejdź na google.pl
Konto NVIDIA	Bezpłatne	Zarejestruj na <a href="https://build.nvidia.com">build.nvidia.com</a>

### ✗ Co NIE jest potrzebne

- ✗ Karta graficzna NVIDIA (model działa w chmurze)
- ✗ Instalacja Pythona (LibreOffice ma wbudowanego)
- ✗ Dodatkowe biblioteki lub pakiety
- ✗ Uprawnienia administratora (opcjonalnie)

## 2. Pobieranie klucza API

**Krok 1:** Otwórz przeglądarkę i wejdź na adres: <https://build.nvidia.com>

**Krok 2:** Kliknij przycisk "**Sign In**" (Zaloguj się) w prawym górnym rogu.

**Krok 3:** Utwórz konto lub zaloguj się przez Google, GitHub lub email firmowy.

**Krok 4:** W wyszukiwarce wpisz: `bielik`

**Krok 5:** Wybierz model: **speakleash/bielik-11b-v2.6-instruct**

**Krok 6:** Kliknij przycisk "**Get API Key**" (Pobierz klucz API)

**Krok 7:** Skopiuj klucz. Wygląda tak: `nvapi-xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx`

**⚠ Ważne:** Klucz zaczyna się od `nvapi-`. Zachowaj go — będzie potrzebny w następnym kroku.

## 3. Instalacja makra

### Gdzie umieścić plik

**Krok 1:** Pobierz plik `polonista.py` ze strony [neuroatypowi.org](http://neuroatypowi.org)

**Krok 2:** Otwórz Eksplorator plików (Windows + E)

**Krok 3:** W pasku adresu wklej: `%APPDATA%\LibreOffice\4\user\Scripts\python`

**Krok 4:** Naciśnij Enter

**Krok 5:** Skopiuj plik `polonista.py` do tego folderu

**Krok 6:** Zamknij WSZYSTKIE okna LibreOffice

**Krok 7:** Otwórz LibreOffice Writer ponownie

 **Uwaga:** Jeśli folder `python` nie istnieje, utwórz go: Kliknij prawym przyciskiem → Nowy → Folder → Nazwij: `python`

## 4. Konfiguracja klucza API

### Edycja pliku `polonista.py`

**Krok 1:** Otwórz folder z makrem (patrz sekcja 3)

**Krok 2:** Kliknij prawym przyciskiem na plik `polonista.py`

**Krok 3:** Wybierz: **Otwórz za pomocą → Notatnik**

**Krok 4:** Znajdź linię (około linia 55):

```
return "nvapi-TUTAJ_WKLEJ_SWOJ_KLUCZ_API"
```

**Krok 5:** Zamień `TUTAJ_WKLEJ_SWOJ_KLUCZ_API` na swój prawdziwy klucz

**Krok 6:** Zapisz plik (Ctrl + S)

**Krok 7:** Zamknij Notatnik

**Krok 8:** Uruchom ponownie LibreOffice Writer

## 5. Użytkowanie programu

### Jak upraszczać tekst

**Krok 1:** Otwórz dokument w LibreOffice Writer

**Krok 2:** Zaznacz tekst do uproszczenia (przeciagnij myszką)

**Krok 3:** Menu: **Narzędzia → Makra → Uruchom makro...**

**Krok 4:** W oknie "Wybór makra": Rozwiń **Moje makra → polonista**

**Krok 5:** Wybierz: **RedagujZaznaczenie**

**Krok 6:** Kliknij przycisk: **Uruchom**

**Krok 7:** Poczekaj 3-10 sekund

✓ **Gotowe!** Zaznaczony tekst zostanie zastąpiony uproszczoną wersją.

### Dostępne funkcje (makra)

Funkcja	Opis	Kiedy użyć
<b>RedagujZaznaczenie</b>	Upraszcza zaznaczony tekst	Główna funkcja — używaj najczęściej
<b>PokazInformacje</b>	Wyświetla informacje o programie	Gdy chcesz sprawdzić wersję
<b>SprawdzKonfiguracje</b>	Sprawdza czy klucz API jest poprawny	Przy pierwszym uruchomieniu
<b>TestPolaczenia</b>	Testuje połączenie z serwerem	Gdy coś nie działa

## 6. Rozwiązywanie problemów

### Problem 1: Makro nie jest widoczne w LibreOffice

**Objaw:** W oknie "Uruchom makro" nie ma folderu **polonista**

**Przyczyna:** Plik jest w złym miejscu lub LibreOffice nie odświeżył cache

**Rozwiązańie:**

1. Sprawdź czy plik jest w: %APPDATA%\LibreOffice\4\user\Scripts\python\polonista.py
2. Zamknij WSZYSTKIE okna LibreOffice
3. Poczekaj 10 sekund
4. Otwórz LibreOffice Writer

**Problem 2: Funkcje makra nie są widoczne w polu "Nazwa makra"**

**Objaw:** Folder polonista jest widoczny, ale po prawej stronie lista jest pusta

**Przyczyna:** Błąd składni w pliku Python lub nieprawidłowe kodowanie pliku

**Rozwiązańie:**

1. Pobierz ponownie plik polonista.py ze strony neuroatypowi.org
2. Upewnij się, że plik jest zapisany jako UTF-8
3. Sprawdź czy nie ma polskich znaków w ścieżce do pliku
4. Usuń stary plik i wgraj nowy

**Problem 3: Błąd "Nieprawidłowy klucz API"**

**Objaw:** Po uruchomieniu makra pojawia się komunikat o błędzie klucza

**Rozwiązańie:**

1. Otwórz plik polonista.py w Notatniku
2. Sprawdź czy klucz zaczyna się od nvapi-
3. Sprawdź czy nie ma spacji przed ani po kluczu
4. Sprawdź czy klucz jest w cudzysłowach: "nvapi-xxx"

**Problem 4: Błąd "Limit zapytań"**

**Objaw:** Komunikat "[BLAD] Limit zapytań (429)"

**Rozwiązańie:** Poczekaj 1-2 minuty i spróbuj ponownie. Upraszczaj mniejsze fragmenty tekstu.

**Problem 5: Brak połączenia z internetem**

**Objaw:** Komunikat "[BLAD] Brak połączenia z internetem"

**Rozwiązańie:**

1. Sprawdź połączenie internetowe (otwórz google.pl w przeglądarce)
2. Sprawdź czy firewall nie blokuje LibreOffice
3. Wyłącz VPN jeśli jest włączony

## 7. Proponowane zmiany dla projektu localwriter

### Przegląd projektu

Projekt **localwriter** ([github.com/balisujohn/localwriter](https://github.com/balisujohn/localwriter)) to rozszerzenie LibreOffice Writer umożliwiające lokalne przetwarzanie tekstu przez modele AI.

### Zidentyfikowane błędy i proponowane poprawki

Błąd	Kod oryginalny	Kod poprawiony
Parsowanie odpowiedzi	<code>result['choices'][0]['message']['content']</code>	<code>result['choices'][0]['message']['content']</code>
Inicjalizacja zmiennej	<code>redacted_paragraphs =</code>	<code>redacted_paragraphs = []</code>
Brak walidacji klucza	Brak sprawdzenia	Funkcja <code>validate_api_key()</code>
Ciche błędy	<code>except: return ""</code>	<code>except Exception as e: show_error(e)</code>

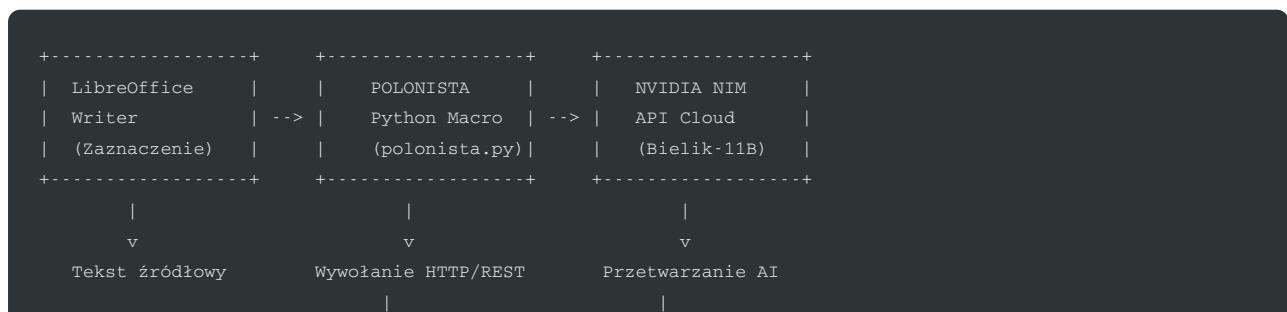
### Proponowana integracja z NVIDIA NIM

```
# Nowy backend dla localwriter: nvidia_nim_backend.py
NVIDIA_NIM_CONFIG = {
    "endpoint": "https://integrate.api.nvidia.com/v1/chat/completions",
    "model": "speakeash/bielik-11b-v2.6-instruct",
    "rate_limit": 40, # requests per minute
    "max_tokens": 2048
}

def create_nvidia_payload(text, system_prompt, temperature=0.3):
    return {
        "model": NVIDIA_NIM_CONFIG["model"],
        "messages": [
            {"role": "system", "content": system_prompt},
            {"role": "user", "content": text}
        ],
        "temperature": temperature,
        "max_tokens": NVIDIA_NIM_CONFIG["max_tokens"],
        "stream": False
    }
```

## 8. Dokumentacja techniczna

### Architektura POLONISTA



```
+-----+  
|  
v  
Zamiana tekstu w dokumencie
```

## Specyfikacja API NVIDIA NIM

Parametr	Wartość
Endpoint	<a href="https://integrate.api.nvidia.com/v1/chat/completions">https://integrate.api.nvidia.com/v1/chat/completions</a>
Model	speakleash/bielik-11b-v2.6-instruct
Metoda	POST
Content-Type	application/json
Authorization	Bearer {api_key}
Rate Limit	40 żądań/minutę
Max Tokens (output)	2048
Timeout	60 sekund

## Kody błędów HTTP

Kod	Znaczenie	Rozwiążanie
200	Sukces	—
401	Błąd autoryzacji	Sprawdź klucz API
429	Limit zapytań	Poczekaj i spróbuj ponownie
500	Błąd serwera	Spróbuj później

## 9. Bibliografia

Cytowanie APA 7	Łącze
NVIDIA Corporation. (2024). <i>NVIDIA NIM API documentation</i> . <a href="https://docs.nvidia.com/nim/">https://docs.nvidia.com/nim/</a>	<a href="https://docs.nvidia.com/nim/">docs.nvidia.com/nim</a>
SpeakLeash. (2024). <i>Bielik-11B: Polish language model</i> . Hugging Face. <a href="https://huggingface.co/speakleash/bielik-11b-v2.6-instruct">https://huggingface.co/speakleash/bielik-11b-v2.6-instruct</a>	<a href="https://huggingface.co/speakleash">huggingface.co/speakleash</a>
Balis, J. (2024). <i>localwriter: LibreOffice extension for local AI</i> . GitHub. <a href="https://github.com/balisujohn/localwriter">https://github.com/balisujohn/localwriter</a>	<a href="https://github.com/balisujohn/localwriter">github.com/balisujohn/localwriter</a>
The Document Foundation. (2024). <i>LibreOffice Python scripting guide</i> . <a href="https://wiki.documentfoundation.org/Macros/Python_Guide">https://wiki.documentfoundation.org/Macros/Python_Guide</a>	<a href="https://wiki.documentfoundation.org/Macros/Python_Guide">wiki.documentfoundation.org</a>
ISO. (2023). <i>ISO 24495-1:2023 Plain language — Part 1</i> . International Organization for Standardization.	<a href="https://www.iso.org/standard/78907">iso.org/standard/78907</a>

**POLONISTA v2.0** — Stowarzyszenie Zwykłe "Neuroatypowi"

✉ [neuroatypowi.org](http://neuroatypowi.org) | ✉ kontakt@neuroatypowi.org

Licencja: MIT | Data dokumentu: 2025-12-11

*Dokument przygotowany zgodnie z normą ISO 24495-1:2023 (Prosty Język)*