**Analiza dostępnych modeli open source z możliwością lokalnej nauki modelu**

* **LLaMA -model językowy stworzony przez Meta. Model ten jest zaprojektowany do obsługi różnych zadań związanych z przetwarzaniem języka naturalnego (NLP), takich jak generowanie tekstu, tłumaczenie, odpowiedzi na pytania i inne.**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **Dobrze zoptymalizowane, zapewnia świetną jakość wyników przy mniejszych wymaganiach, w porównaniu do konkurencji** | **Nadal wymaga dużych zasobów do działania, co może stanowić wyzwanie dla słabszych urządzeń** |
| **Wiele różnych wariantów do wyboru, pozwalając dopasować model pod swoje potrzeby** | **Słabsze warianty np.7B nie są zbyt wydajne przy złożonych problemach** |
| **Posiada rozbudowaną dokumentację, która jest cały czas rozwijana** | **Ograniczone wsparcie innych języków niż angielski** |
| **„Kreatywność” modelu który potrafi generować różnorodne teksty w formie literackiej np. wiersze, scenariusze, czy nawet utwory muzyczne** |  |
| **Rozbudowana baza danych gwarantuje precyzyjne odpowiedzi na nawet najbardziej złożone i wieloetapowe instrukcje, zapewniając wysoką jakość i trafność wyników** |  |

* **Mistral- zaprojektowany przez Mistral AI.**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **Niższe wymagania GPU niż LLaMA2** | **Stosunkowo mała dokumentacja** |
| **Przystosowany do szybkiego wdrożenia w help deskach i chat botach** | **Nowy model co przekłada się na mniejszą ilość testów.** |
| **Bardzo dobre wyniki w konwersacjach oraz szybkich instrukcjach** | **Gorsza jakość odpowiedzi niż LLaMA2 przy trudnych zagadnieniach** |
|  | **Mniejsza efektywność przy złożonych, wieloetapowych instrukcjach technicznych** |
|  | **Ograniczone wsparcie innych języków niż angielski** |

* **GPT(Neo, Neo-J, Neo-X)**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **Brak ograniczeń licencyjnych (Apache 2.0)** | **Słabe rozumienie kontekstu w porównaniu do powyższych modeli** |
| **GPT J, Neo wymagają minimalnych zasobów GPU** | **Najniższa jakość generowania odpowiedzi przy skomplikowanych zadaniach** |
| **Duża, aktywna społeczność** | **Neo-20B zużywa dużo zasobów GPU bez istotnego wzrostu jakości** |

* **BLOOM- jeden z największych dostępnych open-source modeli językowych, stworzony przez BigScience w 2022.**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **Wielojęzyczny, w porównaniu do większości modeli, gdzie głównie obsługiwany jest język angielski, Bloom obsługuje aż 46 języków(w tym polski)** | **Duże wymagania sprzętowe, przez co może być trudny do wdrożenia** |
| **Wszechstronność, radzi sobie swietnie z różnymi instrukcjami językowymi** | **Ograniczenia użytkowania w zakresie etyki, co może powodować problemy przy niektórych projektach(licencja RAIL)** |

* **Falcon- seria otwartoźródłowych modeli językowych opracowanych przez Technology Innovation Institute (TII)**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **Najwyższa efektywność obliczeniowa spośród innych modeli** | **Bardzo wysokie wymagania sprzętowe** |
| **pełna swoboda użytkowania(licencja Apache 2.0)** | **Mniejsze wsparcie społecznościowe** |
| **Zaprojektowany do użytku przy wielu różnorodnych zadaniach** |  |
| **Wiele języków jest wspieranych** |  |

* **DeepSeek- został stworzony przez DeepSeek AI**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **Dobra efektywność obliczeniowa, ale gorzej zoptymalizowana** | **Słabo rozbudowana dokumentacja i mniejsza społeczność niż u innych modeli** |
| **pełna swoboda użytkowania(licencja Apache 2.0)** | **Nieco słabszy przy „kreatywnym” pisaniu** |
| **Świetny przy generowaniu kodu** | **Niektóre wrażliwe tematy w szczególności te dotyczące Chin mogą być objęte cenzurą** |
|  | **Dane użytkownika są przechowywane w Chinach** |
|  | **Ograniczona dostępność językowa (Głównie chiński i angielski)** |

* **Gemma- model opracowany przez Google**

|  |  |
| --- | --- |
| **Plusy** | **Minusy** |
| **pełna swoboda użytkowania(licencja Apache 2.0)** | **Ograniczona wielojęzyczność w porównaniu do BLOOM czy Mistral** |
| **Dobra optymalizacja** | **Mniejsze wsparcie społecznościowe** |
| **Dobra wydajność podczas generowania tekstu** | **Słabsza efektywność przy cięższych zadaniach** |