

Narzędzie do rozpoznawania języka oparte na bibliotece fastText

Jacek Godzicki
Jakub Jasiński
Mateusz Cholewa

Abstrakt

Projekt miał na celu stworzenie narzędzia do rozpoznawania języka przy użyciu biblioteki fastText. Narzędzie to, oparte na zaawansowanych technikach przetwarzania języka naturalnego, ma na celu automatyczne identyfikowanie języka tekstu wejściowego. Projekt obejmuje etapy od przygotowania danych, poprzez trenowanie modelu zgodnie z wytycznymi z oficjalnego bloga fastText, aż po implementację i testowanie narzędzia. Rezultaty pokazują wysoką skuteczność narzędzia w rozpoznawaniu różnych języków, co czyni je użytecznym w szerokim zakresie zastosowań, od analizy danych po lokalizację oprogramowania.

Wstęp

Cel

Celem projektu było opracowanie narzędzia do automatycznego rozpoznawania języka tekstu, wykorzystującego bibliotekę fastText. FastText, opracowana przez Facebook AI Research (FAIR), jest wydajną i elastyczną biblioteką do analizy reprezentacji słów oraz klasyfikacji tekstu. Nasze narzędzie ma na celu ułatwienie identyfikacji języka w różnych kontekstach, takich jak analiza treści internetowych, przetwarzanie danych tekstowych czy systemy tłumaczeń automatycznych.

Zakres

Realizacja projektu obejmowała następujące etapy:

- Przegląd literatury i narzędzi dostępnych do rozpoznawania języka.
- Przygotowanie i przetwarzanie zbiorów danych.
- Implementację modelu rozpoznawania języka z wykorzystaniem biblioteki fastText.
- Stworzenie graficznego interfejsu użytkownika przy użyciu biblioteki tkinter.
- Testowanie i ocena efektywności narzędzia.

Metodyka

Do realizacji projektu wykorzystano następujące narzędzia i metody:

- Język Python do implementacji i przetwarzania danych.
- Bibliotekę fastText do trenowania modelu rozpoznawania języka.
- Zbiory danych, wykorzystane do uczenia modelu, pobrane z serwisu Tatoeba.
- Techniki przetwarzania języka naturalnego (NLP) do przygotowania danych.
- Bibliotekę tkinter do stworzenia graficznego interfejsu użytkownika.

Część Teoretyczna

W tej części omówione zostały podstawy teoretyczne niezbędne do zrozumienia działania narzędzia.

FastText

FastText to biblioteka open-source stworzona przez Facebook AI Research (FAIR), która umożliwia szybkie i wydajne tworzenie reprezentacji słów oraz klasyfikację tekstu. FastText rozszerza model word2vec, umożliwiając uwzględnianie n-gramów, co pozwala na lepsze reprezentowanie rzadkich i złożonych słów.

Rozpoznawanie języka

Rozpoznawanie języka polega na automatycznej identyfikacji języka tekstu wejściowego. Jest to kluczowy komponent wielu systemów NLP, takich jak tłumaczenia automatyczne, wyszukiwanie informacji czy analiza sentymentu.

Część Praktyczna

Przygotowanie danych

Dane do trenowania modelu zostały pobrane z serwisu Tatoeba, który dostarcza zbiory zdań w różnych językach. Proces przygotowania danych został przeprowadzony zgodnie z wytycznymi z oficjalnego bloga fastText. Kroki obejmowały:

Pobranie danych: Pobranie danych językowych z serwisu Tatoeba w formacie .tar.bz2. Komendy do pobrania i rozpakowania danych:

```
wget http://downloads.tatoeba.org/exports/sentences.tar.bz2
bunzip2 sentences.tar.bz2
tar xvf sentences.tar
```

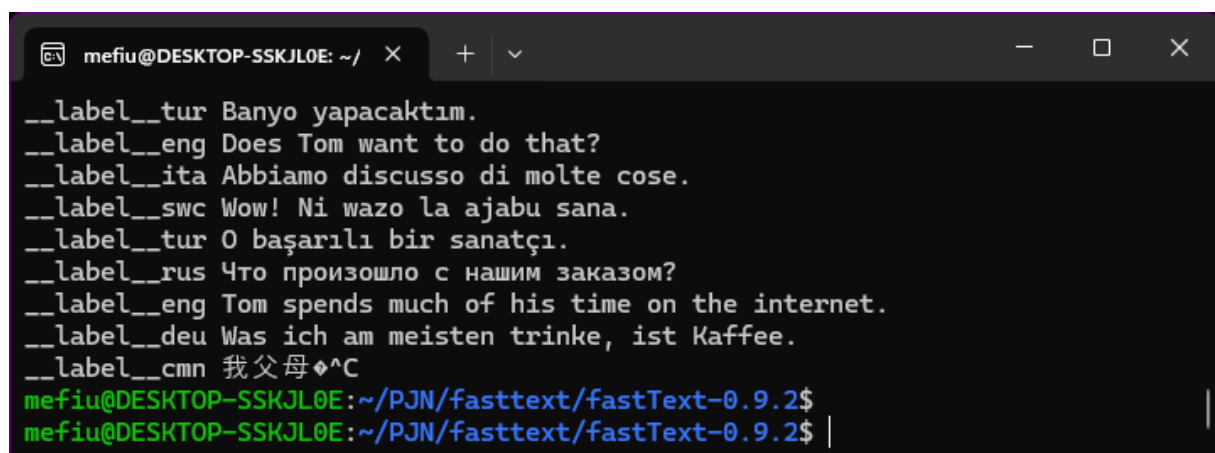
Przygotowanie danych: Dane zawarte w pliku csv zostały przekształcone do formatu wymaganego przez fastText. Każda linia w pliku zawierała zdanie poprzedzone etykietą językową w formacie __label__<label>. Komenda do przekształcenia danych:

```
awk -F"\t" '{print "__label__"$2 "$3"}' < sentences.csv | shuf > all.txt
```

Podział danych: Dane zostały podzielone na zestawy treningowe i walidacyjne, aby umożliwić ocenę modelu po treningu. Komendy do podziału danych:

```
head -n 10000 all.txt > valid.txt
tail -n +10001 all.txt > train.txt
```

Poniżej przedstawiono fragment utworzonego pliku z danymi uczącym "train.txt".



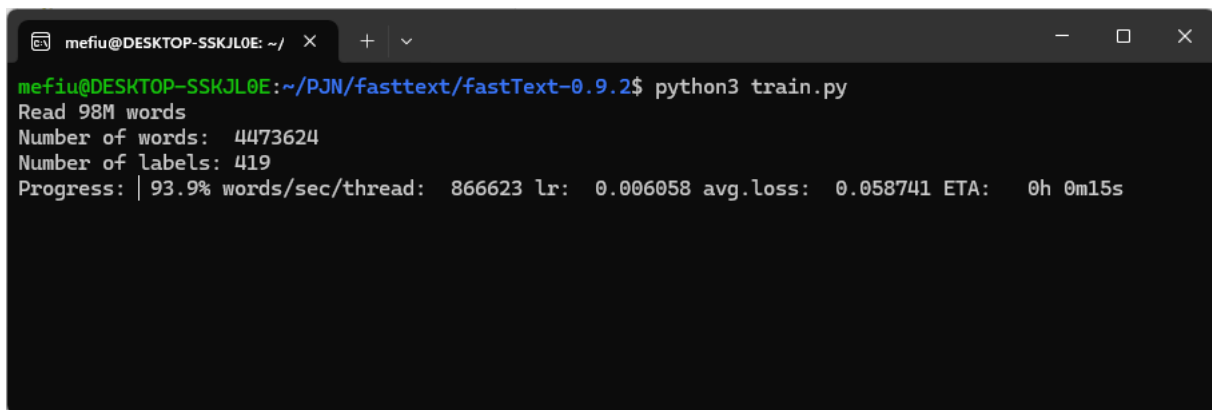
```
mefiu@DESKTOP-SSKJL0E: ~/PJN/fasttext/fastText-0.9.2$
__label__tur Banyo yapacaktım.
__label__eng Does Tom want to do that?
__label__ita Abbiamo discusso di molte cose.
__label__swc Wow! Ni wazo la ajabu sana.
__label__tur O başarılı bir sanatçı.
__label__rus Что произошло с нашим заказом?
__label__eng Tom spends much of his time on the internet.
__label__deu Was ich am meisten trinke, ist Kaffee.
__label__cmn 我父母♠^C
mefiu@DESKTOP-SSKJL0E:~/PJN/fasttext/fastText-0.9.2$
mefiu@DESKTOP-SSKJL0E:~/PJN/fasttext/fastText-0.9.2$ |
```

Trenowanie modelu

Do trenowania modelu użyto biblioteki fastText. Trenowanie modelu przebiegło na przygotowanych danych z oznaczonymi etykietami językowymi.

```
import fasttext

# Trenuj model fastText
model = fasttext.train_supervised(input="data/train.txt", epoch=25, lr=0.1,
wordNgrams=2, bucket=200000, dim=50, loss='hs')
model.save_model("language_recognition_model.bin")
```

A screenshot of a terminal window with a dark background. The window title is 'mefiu@DESKTOP-SSKJL0E: ~/'. The command 'python3 train.py' has been executed. The output shows: 'Read 98M words', 'Number of words: 4473624', 'Number of labels: 419', and 'Progress: | 93.9% words/sec/thread: 866623 lr: 0.006058 avg.loss: 0.058741 ETA: 0h 0m15s'.

```
mefiu@DESKTOP-SSKJL0E:~/PJM/fasttext/fastText-0.9.2$ python3 train.py
Read 98M words
Number of words: 4473624
Number of labels: 419
Progress: | 93.9% words/sec/thread: 866623 lr: 0.006058 avg.loss: 0.058741 ETA: 0h 0m15s
```

Wyjaśnienie parametrów metody train_supervised:

1. input:
 - Opis: Ścieżka do pliku z danymi treningowymi.
2. epoch:
 - Opis: Liczba epok, czyli ile razy cały zbiór danych treningowych zostanie przetworzony podczas trenowania modelu.
 - Znaczenie: Większa liczba epok może prowadzić do lepszego dopasowania modelu, ale także do przeuczenia.
3. lr (learning rate):
 - Opis: Współczynnik uczenia, który określa, jak dużą zmianę wprowadzamy do wag modelu w każdej iteracji.
 - Znaczenie: Wyższy współczynnik uczenia może przyspieszyć konwergencję, ale zbyt wysoki może powodować niestabilność w trenowaniu.
4. wordNgrams:
 - Opis: Maksymalna długość n-gramów słów używanych w modelu.
 - Znaczenie: Umożliwia modelowi uwzględnianie sekwencji słów, co może poprawić dokładność klasyfikacji tekstu.
5. bucket:
 - Opis: Liczba kubełków używanych do haszowania n-gramów słów.
 - Znaczenie: Większa liczba kubełków zmniejsza prawdopodobieństwo kolizji w haszowaniu, co może poprawić dokładność.
6. dim:
 - Opis: Wymiarowość wektorów słów (embedding dimension).

- Znaczenie: Wyższa wymiarowość może prowadzić do bardziej szczegółowych reprezentacji, ale także zwiększa złożoność obliczeniową.

7. loss:

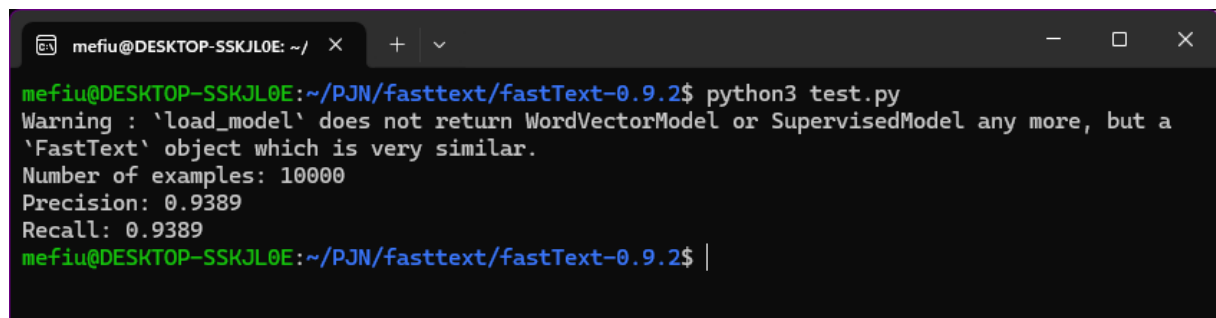
- Opis: Funkcja straty używana do trenowania modelu.
- Opcje:
 - 'softmax': Standardowa funkcja softmax używana do klasyfikacji wieloklasowej.
 - 'hs': Hierarchical softmax, która jest bardziej wydajna dla dużych zbiorów danych.
 - 'ns': Negative sampling, która jest alternatywną metodą trenowania word2vec.
 - 'ova': One-vs-all, która trenuje oddzielny model dla każdej etykiety.

Testowanie modelu

Model został przetestowany na zestawie danych testowych, aby ocenić jego skuteczność. Wyniki testów wskazują na wysoką dokładność rozpoznawania języków.

```
# Testowanie modelu
result = model.test("data/valid.txt")
print("Precision:", result.precision)
print("Recall:", result.recall)
```

Wynik wykonania:



```
mefiu@DESKTOP-SSKJL0E: ~/PJM/fasttext/fastText-0.9.2$ python3 test.py
Warning : 'load_model' does not return WordVectorModel or SupervisedModel any more, but a
'FastText' object which is very similar.
Number of examples: 10000
Precision: 0.9389
Recall: 0.9389
mefiu@DESKTOP-SSKJL0E:~/PJM/fasttext/fastText-0.9.2$ |
```

Trzy zwrócone wartości to:

- **Precyzja (ang. precision):** Jest to stosunek liczby poprawnie przewidzianych pozytywnych przykładów do liczby wszystkich przykładów, które model przewidział jako pozytywne. Innymi słowy, precyzja pokazuje, jak dokładne są przewidywania modelu w kontekście klasyfikacji pozytywnych przykładów. Precyzja jest wyrażona jako liczba z zakresu od 0 do 1.
- **Czułość (ang. recall):** Jest to stosunek liczby poprawnie przewidzianych pozytywnych przykładów do liczby wszystkich rzeczywistych pozytywnych przykładów. Czułość mierzy, jak dobrze model jest w stanie zidentyfikować wszystkie pozytywne przykłady w zbiorze danych. Czułość również jest wyrażona jako liczba z zakresu od 0 do 1.
- **Liczba przykładów (ang. number of examples):** Jest to liczba przykładów użytych do testowania modelu. To po prostu pokazuje, na ilu przykładach model został przetestowany.

Stworzenie interfejsu graficznego

Interfejs graficzny został stworzony przy użyciu biblioteki tkinter. Umożliwia on użytkownikowi wprowadzenie tekstu, którego język ma być rozpoznany, a następnie wyświetlenie wyniku wraz z poziomem pewności.

```

import tkinter as tk
from tkinter import messagebox
import fasttext

# Wczytaj wytrenowany model
model = fasttext.load_model("language_recognition_model.bin")

def detect_language():
    text = entry.get("1.0", tk.END).strip()
    predictions = model.predict(text, k=1)
    language = predictions[0][0].split("__label__")[1]
    confidence = predictions[1][0]
    result_text = f"Language {language} with confidence {confidence}"
    label_result.config(text=result_text)

# Stwórz okno aplikacji
root = tk.Tk()
root.title("Language Recognizer 2000")

entry = tk.Text(root, height=5, width=40)
entry.pack()

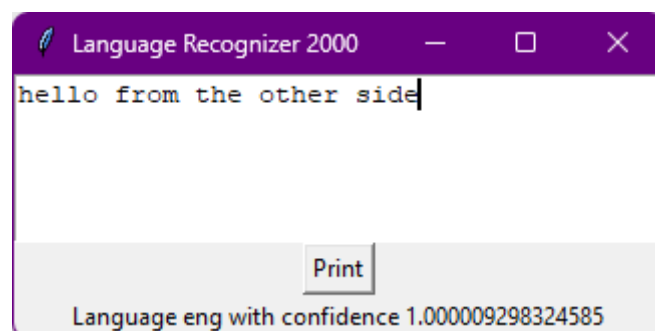
button = tk.Button(root, text="Print", command=detect_language)
button.pack()

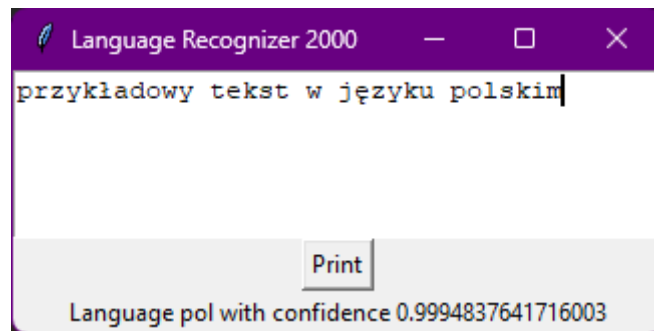
label_result = tk.Label(root, text="")
label_result.pack()

root.mainloop()

```

Uruchomiony program prezentuje się następująco:





Podsumowanie

Celem projektu było stworzenie narzędzia do rozpoznawania języka, co udało się osiągnąć z wysoką dokładnością. Narzędzie może być wykorzystane w różnych aplikacjach, takich jak analiza treści, systemy tłumaczeń automatycznych czy przetwarzanie danych tekstowych. Dalsze prace mogą obejmować rozszerzenie modelu o dodatkowe języki oraz optymalizację pod kątem wydajności.

Bibliografia

- Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of Tricks for Efficient Text Classification. arXiv preprint arXiv:1607.01759.
- Facebook AI Research. (2016). FastText: Library for efficient text classification and representation learning, <https://fasttext.cc>
- Wikipedia. (2024). FastText, <https://en.wikipedia.org/wiki/FastText>
- Tatoeba. (2024). Language Dataset, <https://downloads.tatoeba.org/exports/>