FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KIV/UIR
Klasifikace dokumentů

Tomáš Ott
(A17B0314P)

14. května 2020

Obsah

1 Zadání

Ve zvoleném programovacím jazyce navrhněte a implementujte program, který umožní klasifikovat textové dokumenty do tříd podle jejich obsahu, např. počasí, sport, politika, apod. Při řešení budou splněny následující podmínky:

- Použijte data z českého historického periodika Posel od Čerchova“, která jsou k dispozici na `https://drive.google.com/drive/folders/1mQbBNS43gWFRMHDYdSkQug47cuhPTsHJ?usp=sharing`. V původní podobě jsou data k dispozici na `http://www.portafontium.eu/periodical/posel-od-cerchova-1872?language=cs`.
- Pro vyhodnocení přesnosti implementovaných algoritmů bude NUTNÉ vybrané dokumenty ručně označkovat. Každý student ručně anotuje 10 stran zadaného textu – termín 31.3.2020. Za dodržení termínu obdrží student bonus 10b.
- Přiřazení konkrétních textů jednotlivým studentům spolu s návodem na anotaci a příklady je uloženo spolu s daty na výše uvedené adrese, konkrétně:
  - 0 - vzorová složka (takhle by měl výsledek vypadat)
  - 1, 2, .. , 15, 101, 102, .. - data k anotaci
  - přiřazení souboru studentum.xlsx - určení, jaké soubory má jaký student anotovat. Až budete mít anotaci hotovou, doplňte sem informaci.
  - Anotační příručka - návod, jak články anotovat.
  - Klasifikace dokumentů - kategorie.xlsx - seznam kategorií k anotaci s příklady.
  - sem prace20.pdf - Zadání semestrální práce
- implementujte alespoň tři různé algoritmy (z přednášek i vlastní) pro tvorbu příznaků reprezentující textový dokument.
- implementujte alespoň dva různé klasifikační algoritmy (klasifikace s učitelem):
  - Naivní Bayesův klasifikátor
  - klasifikátor dle vlastní volby

- funkčnost programu bude následující: – spuštění s parametry:
  název klasifikátoru, soubor se seznamem klasifikačních tříd, trénovací množina, testovací množina, parametrizační algoritmus, klasifikační algoritmus,název modelu
  program natrénuje klasifikátor na dané trénovací množině, použije zadaný parametrizační a klasifikační algoritmus, zároveň vyhodnotí úspěšnost klasifikace a natrénovaný model uloží do souboru pro pozdější použití (např. s GUI).– spuštění s jedním parametrem:
  název klasifikátoru, název modelu
  program se spustí s jednoduchým GUI a uloženým klasifikačním modelem. Program umožní klasifikovat dokumenty napsané v GUI pomocí klávesnice (resp. překopírované ze schránky).

- ohodnoťte kvalitu klasifikátoru na dodaných datech, použijte metriku přesnost (accuracy), kde jako správnou klasifikaci uvažujte takovou, kde se klasifikovaná třída nachází mezi anotovanými. Otestujte všechny konfigurace klasifikátorů (tedy celkem 6 výsledků).

Poznámky:

- pro vlastní implementaci není potřeba čekat na dokončení anotace. Pro průběžné testování můžete použít korpus současné češtiny, který je k dispozici na `http://ctdc.kiv.zcu.cz/` (uvažujte pouze první třídu dokumentu podle názvu, tedy např. dokument 05857 zdr ptr eur.txt náleží do třídy zdr“ - zdravotnictví). ”

- další informace, např. dokumentace nebo forma odevzdávání jsou k dispozici na CW pod záložkou Samostatná práce.

2 Analýza zadání

2.1 Algoritmy pro tvorbu příznaků

2.1.1 Bag of words

2.1.2 TF-IDF

2.1.3 N-gram

2.2 Klasifikační algoritmy

2.2.1 Naivní Bayes

2.2.2 K-nejbližších sousedů

3 Implementace

4 Uživatelská příručka

4.1 Nápověda

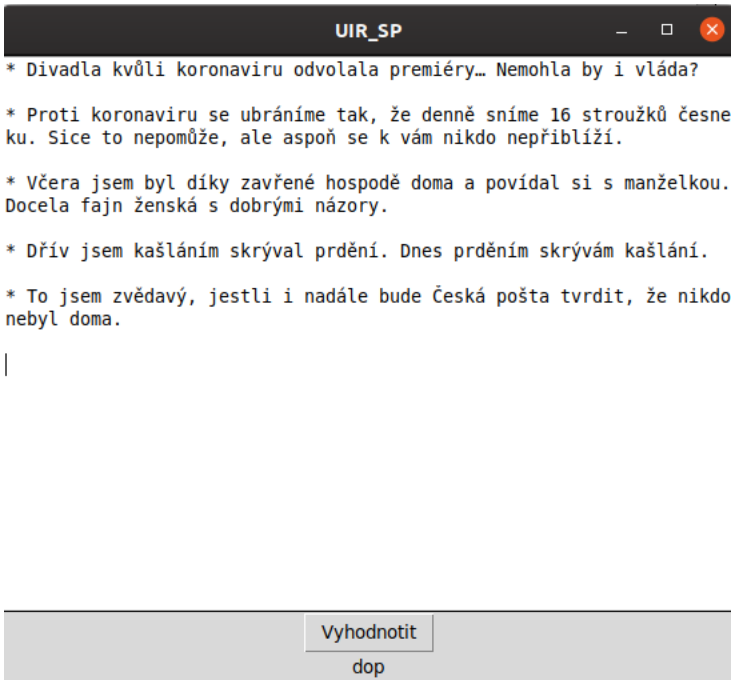
Vypsat nápovědu pro vstupní parametry je možné pomocí příkazu:
"`python Main.py -h`"
Zde jsou vysvětleny všechny parametry, které aplikace přijímá. Mezi ně patří:

- `-c <soubor_se_seznamem_klasifikacnich_trid>` soubor, který obsahuje seznam všech tříd vyskytujících se v jednotlivých dokumentech.
- `-train <trenovaci_mnozina>` množina souborů, která se využije pro naplnění příznakové struktury.
- `-test <testovaci_mnozina>` množina souborů, která budou využity pro ověření funkčnosti klasifikátoru.
- `-p <parametrizacni_algoritmus>` parametrizační algoritmus pomocí kterého budou reprezentovány jednotlivé dokumenty. (Pouze u trénovacího režimu)
- `-k <klasifikacni_algoritmus>` klasifikační algoritmus, který je využit pro klasifikaci dokumentů. (Pouze u trénovacího režimu)
- `<nazev_modelu>` název souboru do kterého je model ukládán a v případě testovacího režimu načítán.

4.2 Trénovací režim

Tento režim je možné spustit zadáním následujícího příkazu:
"`python Main.py -c <soubor_se_seznamem_klasifikacnich_trid>
-train <trenovaci_mnozina> -test <testovaci_mnozina>
-p <parametrizacni_algoritmus> -k <klasifikacni_algoritmus>
<nazev_modelu>`"

Trénovací režim neobsahuje uživatelské rozhraní a všechny jeho výstupy jsou vypisovány do terminálu. Program nejdříve naplní trénovací množinu souborů do zvolené parametrizační struktury. Poté následuje jednotlivé klasifikování testovací množiny souborů, kde jsou všechny výsledky vypsány na obrazovku. Po dokončení druhé fáze následuje vypočtení přesnosti klasifikátoru.



Obrázek 1: Jednoduché uživatelské rozhraní při testovacím režimu

4.3 Testovací režim

Tento režim je možné spustit zadáním následujícího příkazu:
"`python Main.py <nazev_modelu>`"
Aplikace načte předaný model díky kterému vytvoří instance klasifikátoru a struktury příznaků. Tento model je poté využíván pro klasifikaci textu zadaného uživatelem do jednoduchého uživatelského rozhraní zobrazeno na obrázku 1.

5 Závěr