Warning

This page is located in archive. Go to the latest version of this course pages. Go the latest version of this page.

Spam filter - krok 1

Budeme vytvářet funkci, která bude umět načíst údaje ze souborů `!truth.txt` nebo `!prediction.txt` do datové struktury typu `dictionary` .

- Testy [/b211/courses/b4b33rph/cviceni/spam/unit_testing] ke kroku 1: test1_readclassification.zip [/b211/_media/courses/b4b33rph/cviceni/spam/test1_readclassification.zip]

Příprava

1. Práce s datovou strukturou `dictionary` (viz [Pilgrim2009], kapitola 2.7 [http://www.diveintopython3.net/native-datatypes.html#dictionaries], nebo [Wentworth2012], kapitola 20 [http://openbookproject.net/thinkcs/python/english3e/dictionaries.html])
   - Zopakovat: vytvoření prázdného slovníku, přidání páru klíč-hodnota, zjištění hodnoty pro klíč
   - Vyzkoušejte si procházení položek slovníku pomocí metody `items()` :

```
eng_to_cz = {'cat': 'kocka', 'dog': 'pes', 'house': 'dum' }
for eng, cz in eng_to_cz.items():
    print(eng, ',', cz)
```

2. Zopakujte si použití sekce

```
if __name__ == "__main__":
```

   (viz [Pilgrim2009], kapitola 1.10 [http://www.diveintopython3.net/your-first-python-program.html#runningscripts]).
3. Práce se soubory (viz [Pilgrim2009], kapitola 11 [http://www.diveintopython3.net/files.html], nebo [Wentworth2012], kapitola 13 [http://openbookproject.net/thinkcs/python/english3e/files.html] - Ale pozor! Zde autoři zapomínají specifikovat kódování souboru!)

- Otevření a uzavření textového souboru
- Použití příkazu `with`
- Čtení ze souboru (po řádcích)
- Načtení celého obsahu souboru do 1 řetězce

4. Metoda `split()` řetězcových proměnných (viz dokumentace k str.split() [http://docs.python.org/py3k/library/stdtypes.html?highlight=split#str.split])

Načtení klasifikace ze souboru

Úkol:

- V modulu `utils.py` vytvořte funkci `read_classification_from_file` pro načtení klasifikace mailů z textového souboru

K čemu nám to bude:

- Funkci `read_classification_from_file` budeme potřebovat při učení filtru (pokud jej budeme učit) a při hodnocení úspěšnosti filtrů.

Specifikace

Funkce `read_classification_from_file` (v modulu `utils.py`):

Vstupy:	cesta k textovému souboru (v našem případě to budou typicky soubory `!truth.txt` a `!prediction.txt`)
Výstupy:	`dictionary` obsahující pro každý název souboru identifikátor SPAM nebo OK

Funkce načte textový soubor, v němž jsou na řádku vždy 2 řetězce oddělené mezerou,

```
email01.txt OK
email02.msg OK
email03.txt SPAM
email1234.txt OK
...
```

a vytvoří z něj datovou strukturu `dictionary` (na pořadí jednotlivých "řádků" v následujícím výpisu nezáleží):

```
{'email1234.txt': 'OK', 'email03.txt': 'SPAM', 'email02.msg': 'OK', 'email01.txt':
```

Bude-li soubor prázdný, funkce vrátí prázdný slovník.

Zápis klasifikace do souboru

Tip: pravděpodobně se vám bude hodit i inverzní funkce, tedy funkce pro zápis klasifikace uložené ve slovníku do souboru na disku. Toto v tuto chvíli ponecháváme jako dobrovolný DÚ.

courses/b4b33rph/cviceni/spam/krok1.txt · Last modified: 2018/07/17 13:25 (external edit)