# Fake and real news

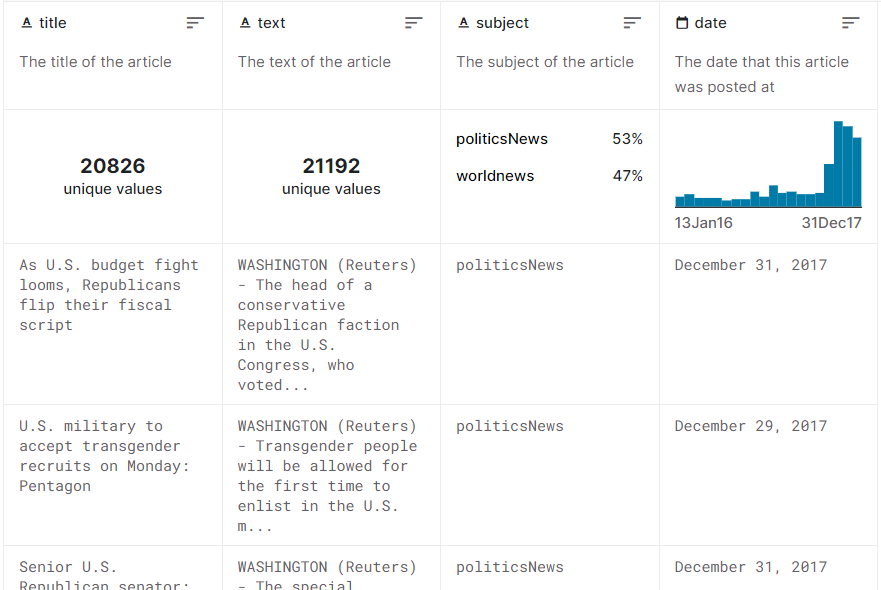
## Abstrakt

V tomto projekte sa zaoberám prácou s datasetom falošných a pravých správ a ich správnej kategorizácii pomocou supervised learning. Medzi iným, porovnávam vplyv veľkosti datasetu na presnosť predikcie, vplyv prítomností titulku a textu na presnosť predikcie a slová ktoré najviac naznačujú pravdivostnú hodnotu správ.

## Úvod

Cieľom tohto projektu je zistiť závislosť presnosti kategorizácie od veľkosti datasetu a jeho ďalšie podobné úpravy. Pri práci budem primárne využívať Python a jeho knižnice.

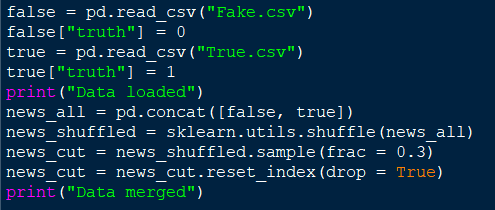
## Dataset

Dataset je získaný zo stránky [Kaggle](https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset/discussion/210551?resource=download&select=True.csv) a skladá sa z dvoch csv súborov, jeden obsahujúci pravdivé správy a druhý falošné. Dataset je tvorený 4 stĺpcami: titulok správy, text správy, téma správy a dátum vydania. Pri mojom projekte využívam len prvé dva stĺpce. Stĺpce téma správy a dátum vydania obsahujú informácie unikátne pre jednu z pravdivostných hodnôt a tým pádom robia klasifikáciu dát triviálnu.

## Metódy a popis kódu

### Načítanie datasetu

Na začiatku treba dáta načítať, na čo som využil knižnicu Pandas. Kedže dáta samé o sebe neobsahujú informáciu o pravdivosti, pridal som k nim ďalší stĺpec vyjadrujúci túto hodnotu, pre falošné správy som priradil 0 a pre pravé 1. Napriek tomu že tento stĺpec neskôr separovaný ako list Y pre samotné učenie. Jeho pripojením som uľahčil miešanie a vzorkovanie dát. Dáta však boli stále rozdelené do dvoch premenných., takže som ich musel zlepiť, premiešať pre ich rovnomerné rozloženie a preindexovať. Z výkonnostných dôvodov som ešte dodatočne zmenšil dataset na tretinu pôvodnej veľkosti.



### Spracovanie datasetu

Takto upravený dataset, konkrétne prvé dva stĺpce, však bolo treba ďalej spracovať na formu spracovateľnú metódami knižnice sklearn. Stĺpce som spracovával oddelene na zachovanie informácie či ide o názov alebo text správy. Na spracovanie som sa inšpiroval jedným z praktických cvičený ktoré sa touto témou zaoberalo.

Ako prvé som odstránil takzvané *stop words*, teda slová, ktoré sa často používajú v jazyku, ale nemajú veľkú hodnotu pre analýzu textu. Zoznam týchto slov som získal z knižnice *nltk.* Všetok text som zároveň zmenil na malé písmená, čo zabráni duplicite slov len následkom ich polohy vo vete. Síce sa tým mierne stráca informačná hodnota dát, keďže písanie uppercasom je typické pre polopravdivé alebo falošné správy, účelom tohto projektu je kategorizácia na základe obsahu a nie formy.

Upravený text bol síce o poznanie kratší a mohol by sa použiť na učenie, avšak slová majúce viacero stále spôsobujú duplicitné skreslenie informácii. Bolo potrebné ich dať do ich základného tvaru, načo som využil metódu *PorterStemmer* z *nltk.stem.porter*. Z textu som následne ešte odstránil nechcené symboly ako bodky, čiarky a zátvorky .

Z takto upraveného textu môžem vytvoriť moju slovnú zásobu, pričom titulok a text ju zdieľajú. Najprv som dáta tokenizoval, využil som na to zase knižnicu *nltk.* Zo zoznamu tokenov som vytvoril ich frekvenčnú tabuľku a vrchných 2500 som použil ako moju slovnú zásobu.

Po získaní slovnej zásoby som už len zmenil názov a text na polia, kde jednotlivé prvky vyjadrujú počty slov v texte. Tieto dva stĺpce som nakoniec zlúčil do jedného zoznamu, takže ich ako vstupné X mám zoznam zoznamov dĺžky 5000 prvkov (2 x dĺžka slovnej zásoby).  
Ako Y som jednoducho použil posledný stĺpec s pravdivostnými hodnotami.

### Trénovanie s rôznymi veľkosťami

V ďalšej časti som trénoval klasifikáciu s rôzne veľkými datasetmi. Dáta som rozdelil pomocou malej funkcie na trénovaciu a testovaciu časť a vytvoril som viacero progresívne menších datasetov. Tieto datasety sú vybraté sekvenčne, čo však neovplyvňuje výsledok, keďže plný dataset už je náhodný a testované datasety majú nulový prienik (teda okrem plného datasetu). Celkovo mám 5 rôznych veľkostí datasetov, pričom veľkosť počet prvkov najväčšieho je 13469 a najmenšieho je 27.

Následne natrénujem SVC pre každý tento dataset. Okrem toho natrénujem dve separátne inštancie pre dataset veľkosti *big*, jednu bez textu a jednu bez názvu správy. Nakoniec využijem *Random Forest Clasifier*, pomocou ktorého môžem jednoducho zistiť ako majú jednotlivé slová vpliv na klasifikovanie správ.