Spamszűrés bayesiánus episztemológiával

Bitai Tamás

ELTE BTK Filozófia Intézet Logika Tanszék

A kéretlen, spam e-mailek kiszűrése a mesterséges intelligencia (*artificial intelligence*, AI), ezen belül a gépi tanulás (*machine learning*, ML), még közelebbről a szövegosztályozás területéhez tartozó probléma. A spamszűrés egyik vezető technológiája a bayesiánus módszer, amelyet Paul Graham írt le 2002-ben [1], és a ma használatos szűrők is, kisebb változtatásokkal, de lényegében ugyanezt az algoritmust alkalmazzák. A következő szakaszban Graham nyomán vázolom a bayesiánus algoritmust.

Bayesiánus spamszűrés

Graham motivációja az volt, hogy az általa korábban írt spamszűrők, amelyek az e-mailek egyedi tulajdonságai alapján működtek (pl. szerepel-e benne a „kattints” szó), nem voltak elég hatékonyak. Ezért egy statisztikai megközelítést javasolt, amelyben az e-mail összes szaváról megnézzük, hogy mekkora arányban fordultak elő korábban a spam leveleinkben, majd ezeket a valószínűség szabályainak megfelelően kombinálva[1] kapunk egy értéket, amely alapján spamnek minősítjük a levelet, ha az a küszöbérték (pl. 0,9) felett van. Az előző példához visszatérve, ha az e-mail teljes szövege „Kattints, hogy nyerj 100 dollárt!”, akkor pl. a következőképpen alakulhat a számítás:

Kattints,	hogy	nyerj	100	dollárt!	
0,99	0,5	0,95	0,85	0,7	→ 0,92

A „kattints” szó magas valószínűséggel spam, a „hogy” semleges (0,5 valószínűségű, tehát ugyanannyiszor fordult elő spam, mint nem spam levelekben) stb. Összességben a spamnek tűnő szavak dominanciája a magas tartományba viszi el a valószínűséget.

A fentiekben vázolt lépéseken kívül a számítást egy szándékos torzítással módosítja Graham: a nem spam levelekbeli előfordulásokat duplán számolja. Ezt azért teszi, mert az általa használt adatokon (Graham a saját levelezését használta teszt adathalmazként) ezzel kevesebb lett a *fals pozitív*, azaz az olyan levél, amit a rendszer spamnek értékelt, de valójában nem az. (A fals pozitív a spamszűrő rendszerek leginkább elkerülendő típusú hibája, mert míg a *fals negatív* levelek a postafiókban megjelenő nem kiszűrt spamek, addig a fals pozitívok a spam mappába kerülő nem spam levelek, amelyek elkerülhetik a felhasználó figyelmét.)

Mint Graham írja, a bayesiánus módszer felismerte a korábban felfedezett spamet indikáló szavakat (mint a „kattints”), ezeken felül pedig még továbbiakat is, amelyekre nem gondolt korábban: ilyen volt pl. az „ff0000”, a piros színnek a HTML-ben használt hexadecimális kódja.

Bayesiánus episztemológia

A bayesiánus spamszűrés algoritmusa valószínűségi feltevésekre alapul. Ebben a szakaszban ezeket a feltevéseket fogjuk részletesebben megvizsgálni. A legalapvetőbb feltevés például, hogy a spamre vonatkozó hiteink mértékét valószínűségekkel írhatjuk le. Általában véve az ilyen és ehhez hasonló elvek filozófiai, közelebbről episztemológiai (ismeretelméleti) tézisek, méghozzá éppen a bayesiánus episztemológia köréből kerülnek ki. A bayesianizmus az episztemológia egyik vezető irányzata, amelynek névadója Thomas Bayes (kb. 1701 – 1761) matematikus és filozófus, akinek a munkásságában megtalálhatók az irányzat korai nyomai. A bayesiánus episztemológia alapelvei a következők:

(B1) Hiteink mértékét valószínűségek írják le.

(B2) A tanulást feltételes valószínűség írja le.

1 Ennek során a 0,5-től legtávolabbi 15 értéken kívül a többit figyelmen kívül hagyjuk.

B1-et szokták *szinkrón tézisnek* is nevezni, mivel a hiteink egy adott időpillanatbeli állapotáról szól; B2-t pedig *diakrón tézisnek*, mert a hiteink időbeli változását jellemzi.

Valószínűségen a formális, matematikai fogalmat értjük, amelyet a Kolmogorov-axiómák definiálnak. Ezek egy *eseménytérre* vonatkoznak, amely egy Ω halmazt és részhalmazainak egy F rendszerét jelenti, melyekre a következő feltételek teljesülnek:

(E1) Ω eleme F-nek.

(E2) Ha A eleme F-nek, akkor $\Omega \setminus A$ (A komplementere) is.

(E3) F elemeinek tetszőleges megszámlálható uniója is eleme F-nek.

F elemeit nevezzük *eseményeknek*.

A valószínűségszámítás alkalmazásaiban a fenti halmazelméleti struktúra gyakran egy logikai struktúrának is megfelel. Például egy kockadobás lehetséges kimeneteleit leíró eseménytérben a dobásokat tartalmazó halmazokat megfeleltethetjük a halmazba tartozást állító mondatoknak. Tehát pl.:

$\{2, 4, 6\}$ → „A dobás páros.”

$\{1, 2, 3, 4\}$ → „A dobás 5-nél kisebb.”

A halmazelméleti műveleteket pedig logikai műveleteknek feleltethetjük meg, pl.:

$\{2, 4, 6\} \cap \{1, 2, 3, 4\} = \{2, 4\}$ (a két halmaz metszete) → „A dobás páros és 5-nél kisebb.”

Ezen megfeleltetés alapján az eseményekre néha nem halmazokként, hanem mondatokként hivatkozunk.

Egyszerű példa eseménytérre a fentebb már említett kockadobásokat leíró algebra. Intuitíve az összes lehetséges kimenetelt leírhatjuk, ha alaphalmaznak az $\Omega = \{1, 2, 3, 4, 5, 6\}$-ot választjuk, F-nek pedig Ω összes részhalmazát.

A spamszűrést leíró eseménytér olyan elemekből állhat, mint:

(spam) „Az e-mail spam.”

(kattints) „Az e-mail tartalmazza a »kattints« szót.”

Valószínűségnek egy olyan, az eseménytéren értelmezett, valós szám értékű P függvényt nevezünk, amelyre a következők teljesülnek:

(K1) $0 \leq P(A) \leq 1$ minden A eseményre.

(K2) $P(\Omega) = 1$.

(K3) $P(\cup_i A_i) = \Sigma_i P(A_i)$, ahol $\{A_i\}_i$ kölcsönösen kizáró események egy megszámlálható halmaza.

A fenti három feltételt nevezik Kolmogorov-axiómáknak, Andrej Kolmogorovról (1903–1987), aki megalapozta a modern valószínűségszámítást.

Az A esemény B eseményre vett *feltételes valószínűségét* a következőképpen definiáljuk:

$P(A \mid B) = P(A \cap B) \,/\, P(B)$.

Egy adott eseményre vett feltételes valószínűség maga is (egy új) valószínűséget ad (teljesíti a Kolmogorov-axiómákat). Így lehetséges a bayesianizmus diakrón tézise szerint feltételes valószínűséggel modellezni a tanulást.

A bayesiánus episztemológia tehát amellett hoz fel érveket, hogy a fenti formális kritériumokkal elemezhető a tanulás. Az egyik leghíresebb ilyen érv a *dutch book* argumentum, amit a következő példával illusztrálhatunk: Képzeljük el, hogy egy lóversenyen négy ló indul, és a fogadóiroda az alábbi táblázat szerint kínál fogadásokat:

Esemény	**Odds**	**Valószínűség**	**Tét**
1. ló nyer	1 : 1	0,5	50 USD
2. ló nyer	3 : 1	0,25	25 USD
3. ló nyer	4 : 1	0,2	20 USD
4. ló nyer	9 : 1	0,1	10 USD

A második oszlopban szerepelnek az oddsoknak megfelelő valószínűségek: tehát pl. a 3 : 1-es oddsnak a 0,25 valószínűség felel meg, mert $0{,}25 \cdot 3 = 0{,}75$ és $0{,}75 : 0{,}25 = 3 : 1$.

Ha a fogadóiroda által kínált oddsoknak megfelelő valószínűségeket helyesnek tartjuk, akkor – szól az érv – amennyiben racionálisak vagyunk, hajlandóak vagyunk fogadni tetszőleges tétben, így pl. a fogadóiroda által kínált tétekben is. De nézzük meg, mi történik, ha megtesszük mind a négy fogadást. A kiadásunk $50 + 25 + 20 + 10 = 105$ dollár lesz. A bevételünk a következő módokon alakulhat:

Esemény	Fizet
1. ló nyer	50 USD + 1 · 50 USD = 100 USD
2. ló nyer	25 USD + 3 · 25 USD = 100 USD
3. ló nyer	20 USD + 4 · 20 USD = 100 USD
4. ló nyer	10 USD + 9 · 20 USD = 100 USD

Vagyis bármelyik ló nyer, a fogadóiroda mindenképpen 100 dollárt fizet. Így azt kaptuk, hogy minden esetben 5 dollárt veszítünk a fogadáson. Az ilyen fogadásokat nevezik dutch booknak.

Vajon mire vezethető vissza, hogy ilyen, biztos veszteséget eredményező fogadást tudott nekünk ajánlani a fogadóiroda? A négy ló győzelme egymást kölcsönösen kizáró események rendszere, és valamelyik biztosan bekövetkezik, így kiadják a teljes eseményteret. Ezért a K3, illetve K2 axiómák szerint a valószínűségeik összegének 1-nek kell lennie. A táblázatban szereplő valószínűségek összege azonban 0,5 + 0,25 + 0,2 + 0,1 = 1,05. Tehát az általunk helyesnek tartott valószínűségek megsértették a Kolmogorov-axiómákat. Ha viszont pl. a 4. ló győzelmére 18 : 1 oddsot fogadtunk volna el, akkor már teljesítettük volna a Kolmogorov-axiómákat, és nem is tudott volna dutch bookot kínálni nekünk a fogadóiroda: pl. ha a számítást egyszerűsítendő 5 dollár tétet tettünk volna erre a lóra, akkor ugyanúgy 100 dollárt fizetett volna a fogadóiroda bármelyik ló győzelme esetén, viszont a mi kiadásunk is csak 100 dollár lett volna. Tehát a példánkban azt láttuk, hogy pontosan akkor viselkedünk racionálisan, ha betartjuk a valószínűségszámítás szabályait.

A bayesianizmus egyik kérdése, hogy a hiteink mértékét kifejező valószínűségeket miként határozzuk meg. Csupán a Kolmogorov-axiómáknak való megfelelés követelménye ugyanis többféle valószínűségi mértéket is lehetővé tesz. Például egy érmedobást értékelhetünk úgy, hogy 0,5 a fej valószínűsége és 0,5 az írásé is, de ha például cinkeltnek gondoljuk az érmét, akkor az is lehet, hogy a fej valószínűségét 0,6-nek értékeljük, az írásét pedig 0,4-nek. Egy lehetséges stratégia a relatív gyakoriságok használata: például ha néhány érmedobás során azt tapasztaljuk, hogy a dobások fele fej, fele írás, akkor gondolhatjuk azt, hogy a következő dobás 0,5 valószínűséggel lesz fej és 0,5 valószínűséggel írás. A relatív gyakoriságokkal való kapcsolatot a dutch book érvhez hasonlóan valószínűségszámítási eredményekkel lehet alátámasztani.

A bayesiánus spamszűrés algoritmusában Graham szintén a gyakoriságok alapján számítja ki a valószínűségeket, de azzal, hogy a nem spam levelekben a gyakoriságokat duplán számolja, eltér a relatív gyakoriságoktól. Ezt az eltérést nem támasztja alá episztemológiai érvekkel, hanem azzal indokolja, hogy a saját e-mailjein így csökkenteni tudta a fals pozitívok számát. A filozófiailag tájékozott olvasóban azonban felmerülhet, hogy ez a cél a bayesianizmus hagyományos keretei között maradva is elérhető, pl. a spamként értékelés küszöbértékének szigorításával. Kérdés továbbá, hogy egyáltalán általános teszt adathalmazon is jobban teljesít-e a Graham-féle torzított valószínűségi algoritmus a tiszta bayesiánus módszernél. A következő szakaszban erre vonatkozó eredményeimet fogom ismertetni.

Bayes vagy spam?

A teszteléshez a bayesiánus spamszűrő algoritmust Pythonban implementáltam, a TensorFlow gépitanulás-csomag segítségével. A kód egy Google Colabon futtatott Jupyter-notebookban érhető el a https://colab.research.google.com/github/tbitai/bayes-or-spam/blob/fe6ebec/bayes_or_spam.ipynb címen.[2]

Teszt adatként a Vangelis Metsis és szerzőtársai 2006-os cikke [2] online változatából elérhető `Enron1` adathalmazt vizsgáltam. Ez az Enron energiaszolgáltató cég elleni tárgyalás során nyilvánosságra hozott, a cég egyik alkalmozottjának postafiókján alapuló adathalmaz, amelyet széles körben használnak spamszűrők összehasonlítására.

Az implementációban a Graham által leírtaktól két helyen tértem el:

1. Az algorimus első lépése a *tokenizálás*, vagyis a szöveges adatok szavakra bontása. Ehhez a TensorFlow `tf.keras.prepocessing.text.text_to_word_sequence` függvényét használtam.

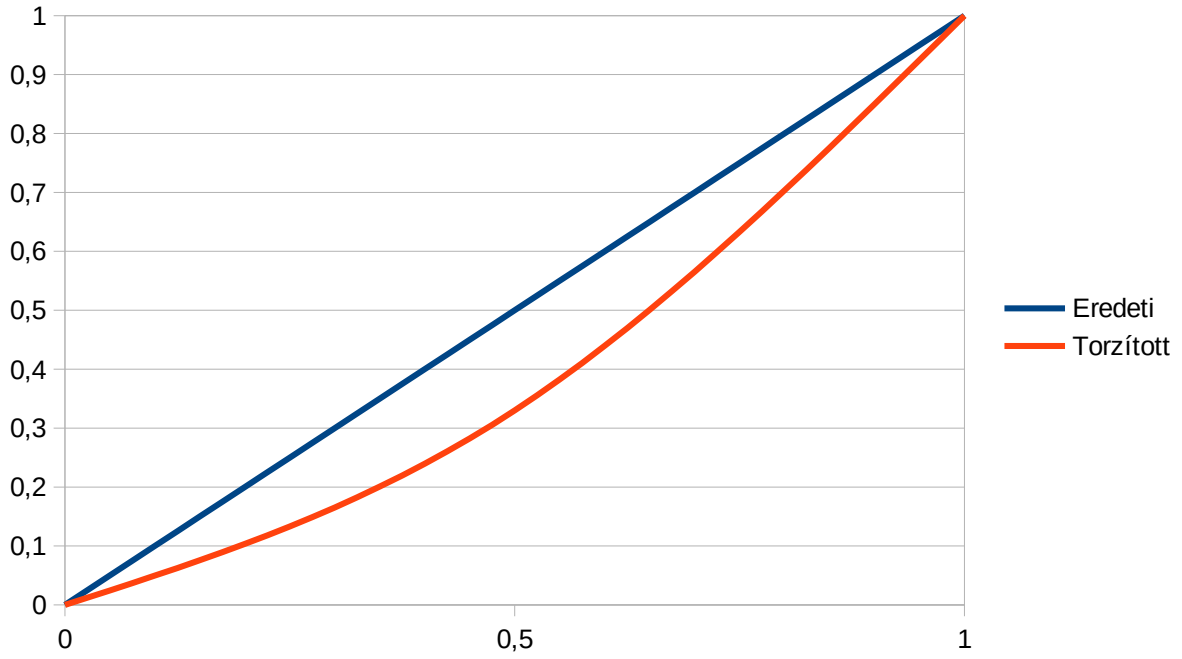
2 A notebookot a későbbiekben tervezem még kiegészíteni, illetve továbbfejleszteni. Ez a változat a Git verziókezelő rendszerben rögzített első commit, amely az első tesztfutás eredményeit tartalmazza.

Graham cikkében egy egyszerű saját megoldást ír le, de megjegyzi, hogy ez fejleszthető, ezért én ezt a modern megoldást választottam.

2. A relatív gyakoriságok Graham-féle formuláját néhány helyen ekvivalens, illetve elhanyagolható különbségű átalakításokkal egyszerűsítettem. (Így a relatív gyakoriságok tiszta matematikai formuláját értem el, a korábban említett, nem spambeli duplán számolásos torzítást kivéve.)

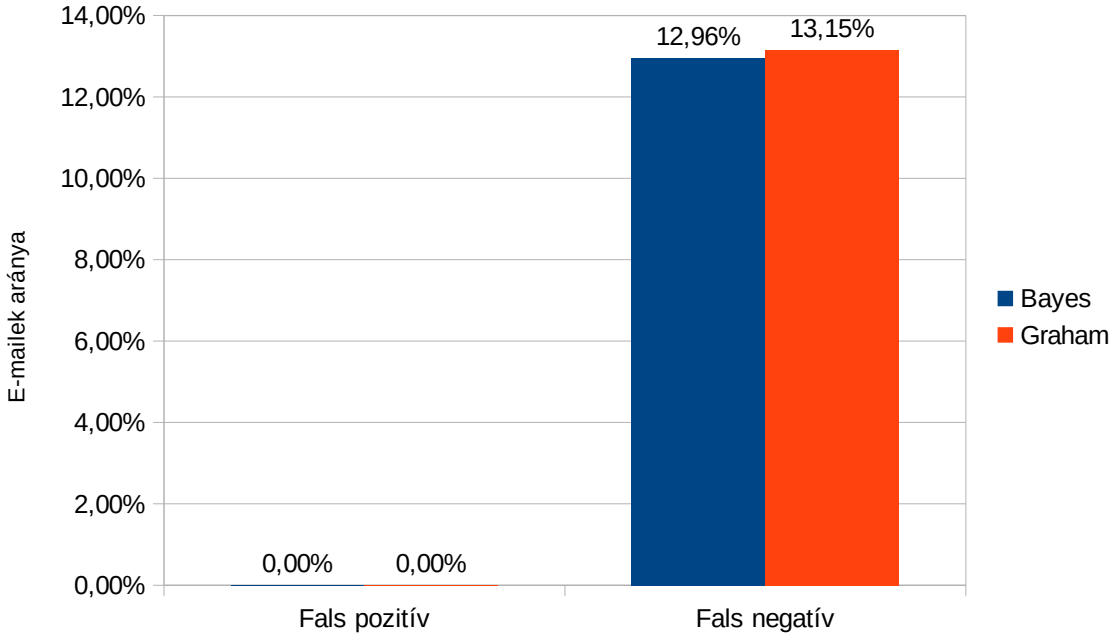
A relatív gyakoriságok formulájába való behelyettesítéssel látható, hogy a nem spam emailek duplán számolásával az eredeti valószínűségek a $p / (2 - p)$ formula szerint torzulnak, ezt a 0 és 1 közötti értékekre az alábbi grafikon ábrázolja:

1. ábra: Torzított valószínűségek



Látható tehát, hogy Graham minden szó valószínűségét kissé csökkentette. Bár ez az ő e-mailjein a fals pozitívok számát csökkentette, általános esetben inkább az lehet az intuíciónk, hogy a fals negatívok számát fogjuk növelni így: ha pl. egy spam csupa olyan szóból áll, amik nagy valószínűséggel spamek, és összességében kicsivel a 0,9-es küszöbérték fölött értékelnénk, akkor a Graham-féle torzítással kicsivel 0,9 alá kerülhetünk, azaz egy ilyen levél éppen átcsúszna a spamszűrőn. Nézzük meg, hogy az `Enron1` adathalmaz igazolja-e az intuíciónkat!

2. ábra: Hibaszázalékok



A fenti diagramon látható, hogy a fals pozitívok száma mind Graham algoritmusával, mind a torzítatlan bayesiánus algoritmussal 0 volt; a fals negatívok aránya pedig Grahamnél 13,15% volt, torzítás nélkül pedig lecsökkent 12,96%-ra.[3] Tehát a torzítatlan bayesiánus algoritmus valóban hatékonyabbnak bizonyult ezen az adathalmazon.

Felmerülhet a kérdés: ez a kb. 0,2%-os hatékonyságnövelés jelentős-e? Ennek a mérlegeléséhez figyelembe kell vennünk, hogy a spamszűrő technológiák már korábban is nagyon hatékonyak voltak, Graham idézett cikkében így ír:

„A spamek *utolsó néhány százalékának* [kiemelés tőlem] felismerése nagyon nehézzé vált".

Egy másik példa a Google 2019-es közleménye [3], amelyben bejelentették a TensorFlow-ban megvalósított gépi tanuló spamszűrők bevezetését a Gmailben, itt Neil Kumaran ezt írja:

„[A Gmail filterei] a spam *több mint 99,9%-ának* blokkolásában segítenek."

(A fordítások a sajátjaim.) Láthatjuk tehát, hogy a spamszűrés területén a tizedszázalékos nagyságrendű eredmények is jónak számítanak – egy ilyen eredménynek tűnik a torzítás kivétele a bayesiánus spamszűrésből, a bayesiánus episztemológia szellemében.

Irodalomjegyzék

[1] Paul Graham: *A Plan for Spam* (2002) http://paulgraham.com/spam.html

[2] Vangelis Metsis, Ion Androutsopoulos, Georgios Paliouras: *Spam Filtering with Naive Bayes – Which Naive Bayes?* (2006) http://www2.aueb.gr/users/ion/publications.html

[3] Neil Kumaran: *Spam does not bring us joy – ridding Gmail of 100 million more spam messages with TensorFlow* (2019) https://cloud.google.com/blog/products/g-suite/ridding-gmail-of-100-million-more-spam-messages-with-tensorflow

3 A számok a teszt első futásakor kapott eredményeket tükrözik. Mivel a kód randomizálást is tartalmaz, ezért későbbi futtatásokra kissé eltérő eredmények adódhatnak.