ISSN 2181-922X

UZBEKISTAN
LANGUAGE & CULTURE

O‘ZBEKISTON:
TIL VA MADANIYAT

AMALIY FILOLOGIYA MASALALARI

2022 Vol. 2 (5)

www.aphil.tssuull.uz

ISSN 2181-922X

O‘ZBEKISTON

TIL VA MADANIYAT

AMALIY FILOLOGIYA MASALALARI

2022 Vol. 2 (5)

www.aphil.tsuull.uz

Alisher Navoiy nomidagi Toshkent davlat o‘zbek tili va adabiyoti universiteti

Bosh muharrir: Saodat Muhamedova

Bosh muharrir o‘rinbosari: Botir Elov

Mas’ul kotib: Xurshida Kadirova

Tahrir kengashi

Aynur O‘zjan (Turkiya), Baydemir Husayn (Turkiya), Alfiya Yusupova (Rossiya), Luiza Samsitova (Rossiya), Almaz Ulvi (Ozarbayjon), Abdulhay Sobirov, Muyassar Saparniyazova, Manzura Abjalova, Nargiza Musulmonova, Yekaterina Shirinova, Shoira Isayeva, Oqila Turaqulova, Ikrom Islomov, Munira Shodmonova, Oqila Abdullayeva, Dilrabo Elova.

Jurnal haqida ma’lumot

“O‘zbekiston: til va madaniyat. Amaliy filologiya masalalari” seriyasi - Oliy attestatsiya komissiyasi ilmiy nashrlar ro‘yxatidagi “O‘zbekiston: til va madaniyat” akademik jurnalining ilovasi hisoblanib, unda professor-o‘qituvchilar, doktorantlar, stajor-tadqiqotchilar, mustaqil izlanuvchilar, magistrantlarning amaliy tilshunoslik, amaliy adabiyotshunoslik, kompyuter lingvistikasi, o‘zbek tilini davlat tili va xorijiy til sifatida o‘qitish, noshirlik ishi kabi sohalarga oid tadqiqotlari nashr qilinadi.

Jurnal ilovasi bir yilda ikki marta chop etiladi.

O‘zbek, rus va ingliz tillaridagi, shuningdek, boshqa turkiy tillarda yozilgan maqolalar qabul qilinadi.

Jurnalda kitoblarga yozilgan taqrizlar, adabiyotlar sharhi, konferensiyalar hisobotlari va tadqiqot loyihalari natijalari ham e’lon qilinadi.

Mualliflar fikri tahririyat nuqtayi nazaridan farq qilishi mumkin.

“O‘zbekiston: til va madaniyat. Amaliy filologiya masalalalari” seriyasi 2022-yildan chiqa boshlagan.

Alisher Navoiy nomidagi Toshkent davlat o‘zbek tili va adabiyoti universiteti.

O‘zbekiston, Toshkent, Yakkasaroy tumani, Yusuf Xos Hojib ko‘chasi, 103.

Email: aphil@tsuull.uz
Website: http://www.aphil.tsuull.uz

MUNDARIJA

Amaliy filologiya

O‘zbek tilini xorijiy til sifatida o‘qitish

Lingvokulturologiya

Tabiiy tilni qayta ishlash (NLP)

Korpus lingvistikasi

ISSN 2181-922X 35—50

TABIIY TILNI QAYTA ISHLASH (NLP)

Tabiiy tilni qayta ishlashda bag of words algoritmidan foydalanish

Botir Elov[1]
Nizomaddin Xudayberganov[2]
Zilola Xusainova[3]

Annotatsiya:

So‘zlar sumkasi modeli - mashinali o‘rganish algoritmlari tomonidan qayta ishlash lozim bo‘lgan matnning raqamli ko‘rinishi. Bag Of Words (BoW) modellashtirish algoritmidan foydalanib, matnni raqamli matritsalarga aylantirish va qayta ishlash mumkin. So‘zlar sumkasi (BoW) - so‘zning hujjatdagi statistikasini hisoblaydigan algoritm. BoW algoritmidan hujjatlarni o‘zaro solishtirish, qidiruv tizimlarida ma’lumotlarni izlash, hujjatlarni tasniflash va tematik modellashtirish kabi NLP ilovalarida foydalaniladi. Ushbu maqolada o‘zbek tilidagi matnlarni BoW alrgoritmi vositasida raqamli shaklga o‘tkazish usullari tahlil qilinadi.

Kalit so‘zlar: *BoW, Bag of words, so‘zlar jamlanmasi, so‘z vektori, token, BoW algoritmi, TF-IDF usuli*.

Kirish

So‘zlar jamlanmasi (“sumkasi”) **(Bag of words**, **BoW**) - bu tabiiy tilni qayta ishlashda matnni modellashtirish (raqamlashtirish) usuli. Ushbu maqolada BoW algoritmining (usuli) qo‘llanilish

[1]*Elov Botir Boltayevich* – texnika fanlari bo‘yicha falsafa doktori (PhD), dotsent. Alisher Navoiy nomidagi Toshkent davlat o‘zbek tili va adabiyoti universiteti Kompyuter lingvistikasi va raqamli texnologiyalar kafedrasi mudiri.
E-pochta: elov@navoiy-uni.uz
ORCID: 0000-0001-5032-6648
[2]*Xudayberganov Nizomaddin Uktamboy o‘g‘li* – Alisher Navoiy nomidagi Toshkent davlat o‘zbek tili va adabiyoti universiteti Kompyuter lingvistikasi va raqamli texnologiyalar kafedrasi o‘qituvchisi.
E-pochta: nizomaddin@navoiy-uni.uz
ORCID: 0000-0002-6213-3015
[3]*Xusainova Zilola Yuldashevna* – Alisher Navoiy nomidagi Toshkent davlat o‘zbek tili va adabiyoti universiteti Kompyuter lingvistikasi va raqamli texnologiyalar kafedrasi o‘qituvchisi.
E-pochta: xusainovazilola@navoiy-uni.uz
ORCID: 0000-0003-4357-7515
Iqtibos uchun: Elov B., Xudayberganov N., Husainova Z. 2022. “Tabiiy tilni qayta ishlashda bag of words alogoritmidan foydalanish”. *O‘zbekiston: til va madaniyat. Amaliy filologiya.* 2 (5): 35-50.

sohasi, konseptsiyasi haqida fikr-mulohaza yuritiladi va o‘zbek tilidagi matnlar uchun Python tilidagi tatbig‘i keltiriladi. Axborot tizimlaridagi katta hajmdagi matnli ma’lumotlarni NLP vositalari yordamida qayta ishlash orqali muhim qarorlar qabul qilinadi. Ushbu katta hajmdagi ma’lumotlarni *tushunish* va muayyan *qarorlarni qabul qilish* uchun ularni **raqamli shaklga olib kelish** kerak [Rudkowsky, Haselmayer, Wastian, Jenny, Emrich, Sedlmair, 2018, 141; Zhang, Jin, Zhou, 2010, 44]. Tabiiy tilni qayta ishlash vositalari usbu vazifani amalga oshirishga yordam beradi.

BoW algoritmi yordamida matnli ma’lumotlardan zarur xususiyatlar ajratib olinadi. Ushbu yondashuv hujjatlardan xususiyatlarni olishning *oddiy va moslashuvchan* usulidir. BoW - bu hujjatdagi so‘zlarning paydo bo‘lishini tavsiflovchi matnning raqamli ko‘rinishi. BoW usulida faqat matndagi so‘zlar soni (statistikasi) aniqlanadi. Biroq grammatik tafsilotlar va so‘z tartibi e’tiborsiz qoldiriladi. Usulning so‘zlar “sumkasi” deb atalishi hujjatdagi so‘zlarning *tartibi yoki tuzilishi* haqidagi har qanday ma’lumotning tashlab yuborilishiga asoslangan [Yan, Li, Gu, Yang, 2020, 82642].

Matn bilan bog‘liq eng katta muammolardan biri shundaki, uning tartibsiz va strukturlanmaganligida. Mashinali o‘rganish algoritmlari esa sturkturlangan va aniq belgilangan qat’iy uzunlikdagi ma’lumotlarni qayta ishlashga asoslangan. BoW algoritmidan foydalanib, o‘zgaruvchan uzunlikdagi matnlarni fiksirlangan uzunlikdagi matn (**vektor**)ga aylantiriladi [Rudkowsky, Haselmayer, Wastian, Jenny, Emrich, Sedlmair, 2018,143; Yan D., Li K., Gu S., Yang L, 2020, 82645; Qiu D., Jiang H., Chen S., 2020, 6]. Mashinali o‘rganish modellarida raqamlashtirilmagan matnli ma’lumotlar qayta ishlanmaydi. Shu sababli, matnlarni raqamli ko‘rinishga olib kelish lozim. BoW algoritmidan foydalanib, matnga ekvivalent raqamlar vektori hosil qilinadi. BoW usuli orqali matnni vektorga aylantirish jarayonini o‘zbek tilidagi gaplar misolida ko‘rib chiqamiz:

1-gap: *“Adirlar ham bahorda lola bilan go‘zal, chunki lola – bahorning erka guli”*

2-gap: *“Lola ham shifokorlik kasbini tanladi”*

1-jadval. Gaplarni tokenlarga ajratish

1-gap	2-gap
Adirlar	Lola
Ham	ham
bahorda	shifokorlik
lola	kasbini
bilan	tanladi

go‘zal	
,	
chunki	
lola	
–	
bahorning	
erka	
guli	

1-qadam: Yuqoridagi matndagi barcha so‘zlarni ko‘rib chiqib, umumiy so‘zlardan iborat lug‘atni shakllantiramiz.

- Adirlar
- ham
- bahorda
- lola
- bilan
- go‘zal
- ,
- chunki
- –
- bahorning
- erka
- guli
- Lola
- shifokorlik
- kasbini
- tanladi

Hosil qilingan lug‘atda **“Lola”** va **“lola”** so‘zlari bir xil emas. Chunki ular gaplarda turli shaklda kelgan va shuning uchun takrorlanmaydi. Shuningdek, so‘zlar ro‘yxatidan vergul: **“,”** ham o‘rin olgan. Berilgan ikkita gapdan hosil qilingan lug‘atda **14 ta** token mavjud bo‘lib, har bir tokenni baholash uchun vektorda bitta pozitsiyaga ega bo‘lgan 14 uzunlikdagi hujjat vektoridan foydalanish mumkin. Matnga mos vektorni hosil qilishda har bir so‘zning mavjudligini qayd etishda **1** yoki **0** qiymatlaridan (chastota) foydalaniladi. 1-gapga mos vektor quyidagicha hosil qilinadi:

2-jadval. 1-gapga mos so‘zlar chastotasi

So‘z	Chastota
Adirlar	1
ham	1
bahorda	1

lola	1
bilan	1
go‘zal	1
,	1
chunki	1
–	1
bahorning	1
erka	1
guli	1
Lola	0
shifokorlik	0
kasbini	0
tanladi	0

1-gapdagi chastotalardan foydalanib quyidagi vektorni hosil qilamiz:

[1,1,1,1,1,1,1,1,1,1,1,0,0,0,0]

2-gapga mos vektor quyidagicha hosil qilinadi:

3-jadval. 2-gapga mos so‘zlar chastotasi

So‘z	Chastota
Adirlar	0
ham	1
bahorda	0
bola	0
bilan	0
go‘zal	0
,	0
chunki	0
–	0
bahorning	0
erka	0
guli	0
Lola	1
shifokor	1
kasbini	1
tanladi	1

Xuddi shunday, yuqoridagi chastotalardan 2-gapga mos vektorni hosil qilamiz:

[0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,1]

Berilgan ikkita so‘zga mos vektorni quyidagicha aniqlaymiz:

	dir lar	ham	Bahor da	lola	bilan	go‘zal	,	chun ki	–	bahor ning	erka	guli	Lola	shifo korlik	kasbini	tan ladi
1-gap	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
2-gap	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Yuqoridagi misolda BoW usulining eng yaxshi jihatlari o‘z aksini topmagan. Chunki, **“Lola”** va “**lola”** so‘zlari bir xil ma’noga ega bo‘lsa-da, ikki marta qayd etilgan. Shuningdek, hech qanday ma’lumotni bildirmaydigan vergul “,”, “–” belgisi ham lug‘at tarkibiga kiritilgan va tahlil jarayonida ishtirok etgan. BoW usuliga ba’zi o‘zgarishlarni amalga oshirish orqali samaradorlikni oshirishni ko‘rib chiqamiz.

Boshlang‘ich ishlov berish

1-gap: *“Adirlar ham bahorda lola bilan go‘zal, chunki lola – bahorning erka guli”*

2-gap: *“Lola ham shifokorlik kasbini tanladi”*

1-qadam: *Berilgan gaplarni kichik harf (registr)ga aylantirish.*

2-qadam: *Matndan maxsus belgilar va nomuhim so‘zlar (stopwords)ni olib tashlash.* Stopwords – *“albatta”, “ba’zi”, “bilan”, “ham”, “chunki”* kabi matnda muhim ahamiyat kasb etmagan so‘zlar. Yuqoridagi amallar bajarilganidan so‘ng, berilgan gaplarda quyidagi o‘zgartirishlar amalga oshiriladi:

1-gap: *“adirlar bahorda lola go‘zal lola bahorning erka guli”*

2-gap: *“lola shifokorlik kasbini tanladi”*

Yuqoridagi gaplar ma’noga ega bo‘lmasa-da, gap mazmuniga ta’sir qiluvchi maksimal ma’lumotlarga ega so‘zlardan iborat.

3-qadam: *Berilgan gaplardagi barcha so‘zlardan iborat lug‘atni shakllantirish*.

- adirlar
- bahorda
- lola
- go‘zal
- bahorning
- erka
- guli
- shifokorlik
- kasbini
- tanladi

Hosil qilingan lug‘atda endilikda **10** ta so‘z mavjudligi sababli, har bir so‘zni baholash uchun vektorda bitta o‘ringa ega bo‘lgan 10 ta qat’iy uzunlikdagi hujjat vektoridan foydalanish mumkin. 1-gapga

mos vektor quyidagicha hosil qilinadi:

3-jadval. 1-gapga mos so‘zlar chastotasi

So‘z	Chastota
Adirlar	1
bahorda	1
lola	2
go‘zal	1
bahorning	1
erka	1
guli	1
shifokorlik	0
kasbini	0
tanladi	0

1-gapdagi chastotalardan foydalanib quyidagi vektorni hosil qilamiz:

[1,1,2,1,1,1,1,0,0,0]

2-gapga mos vektor quyidagicha hosil qilinadi:

4-jadval. 2-gapga mos so‘zlar chastotasi

So‘z	Chastota
adirlar	0
bahorda	0
lola	1
go‘zal	0
bahorning	0
erka	0
guli	0
shifokorlik	1
kasbini	1
tanladi	1

2-gapdagi chastotalardan foydalanib quyidagi vektorni hosil qilamiz:

[0,0,1,0,0,0,0,1,1,1]

Berilgan ikkita so‘zga mos vektorni quyidagicha aniqlaymiz:

	Adirlar	**bahorda**	**lola**	**go‘zal**	**bahorning**	**erka**	**guli**	**shifokorlik**	**kasbini**	**tanladi**
1-gap	1	1	2	1	1	1	1	0	0	0
2-gap	0	0	1	0	0	0	0	1	1	1

Mashinali o‘rganishda ishlatiladigan ma’lumotlar to‘plamlari juda katta hajmda bo‘lib, bir necha ming yoki hatto millionlab so‘zlardan iborat lug‘atni o‘z ichiga olishi mumkin. Shunday qilib, so‘zlar jamlanmasidan foydalanishdan oldin matnni boshlang‘ich qayta ishlash lozim. BoW usuli samaradorligini oshirishi mumkin bo‘lgan turli xil qayta ishlash bosqichlari mavjud bo‘lib, ularning ba’zilari ushbu maqolada batafsil keltiriladi.

Yuqoridagi misollarda BoW vektorini hosil qilish uchun lug‘atdagi barcha so‘zlardan foydalandik. Bu BoW modelini amalga tatbiq qilishda bir qator murakkkabliklarni yuzaga keltiradi. Amalda vektorni yaratish uchun lug‘atdan faqat bir nechta so‘zlar (eng ko‘p uchragan so‘zlar) ishlatiladi.

BoW usulini Pythonda amalga oshirish

BoW algoritmini Python tilida quyidagicha tatbiq qilamiz:

```
def vectorize(tokens):
    vector=[]
    for w in filtered_vocab:
        vector.append(tokens.count(w))
    return vector

def unique(sequence):
    seen = set()
    return [x for x in sequence if not (x in seen or seen.
add(x))]

# Nomuhim so‘zlar ro‘yxatini yaratish
stopwords=[“ham”,”bilan”,”chunki”,…]

# Maxsus belgilar ro‘yxati
special_char=[“,”,”:”,” “,”;”,”.”,”?”,”-”]

# Korpusdagi gaplarni aniqlash
string1=”Adirlar ham bahorda lola bilan go‘zal, chunki lola -
bahorning erka guli”
string2=”Lola ham shifokor bo‘lib ishlaydi”

# so‘zni kichik harfga aylantirish
string1=string1.lower()
string2=string2.lower()

# gaplarni tokenlarga ajratish
```

```
tokens1=string1.split()
tokens2=string2.split()
print(tokens1)
print(tokens2)

# lug‘at ro‘yxatini tuzish
vocab=unique(tokens1+tokens2)
print(vocab)

# lug‘atlar ro‘yxatini filtrlash
filtered_vocab=[]
for w in vocab:
   if w not in stopwords and w not in special_char:
      filtered_vocab.append(w)
print(filtered_vocab)

# gaplarni vektorlarga aylantirish
vector1=vectorize(tokens1)
print(vector1)
vector2=vectorize(tokens2)
print(vector2)
```

_____

[‘adirlar’, ‘ham’, ‘bahorda’, ‘lola’, ‘bilan’, “go‘zal”, ‘,, ‘chunki’, ‘lola’, ‘-’, ‘bahorning‘, ‘erka’, ‘guli’]

[‘lola’, ‘ham’, ‘shifokor’, “bo‘lib”, ‘ishlaydi’]

[‘adirlar’, ‘ham’, ‘bahorda’, ‘lola’, ‘bilan’, “go‘zal”, ‘,, ‘chunki’, ‘-’, ‘bahorning‘, ‘erka’, ‘guli’, ‘shifokor’, “bo‘lib”, ‘ishlaydi’]

[‘adirlar’, ‘bahorda’, ‘lola’, “go‘zal”, ‘bahorning‘, ‘erka’, ‘guli’, ‘shifokor’, “bo‘lib”, ‘ishlaydi’]

[1, 1, 2, 1, 1, 1, 1, 0, 0, 0]

[0, 0, 1, 0, 0, 0, 0, 1, 1, 1]

Sklearn yordamida BoWni ishlab chiqish.

Yuqoridagi BoW modelini Python yordamida osongina amalga oshirish uchun **Sklearn** kutubxonasidagi **Count Vectorizer ()** funksiyasidan foydalanish mumkin.

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer

sentence_1=”Adirlar ham bahorda lola bilan gozal , chunki
```

```
lola - bahorning erka guli”
        sentence_2=”Lola ham shifokor”

        # ngram_range=(2,2) bigramlardan foydalanish
        CountVec = CountVectorizer(ngram_range=(1,1), stop_
words=[“ham”,”bilan”,”chunki”])

        # transformmatsiya
        Count_data = CountVec.fit_
transform([sentence_1,sentence_2])

        # dataframeni shakllantirish
        cv_dataframe=pd.DataFrame(Count_data.
toarray(),columns=CountVec.get_feature_names())
        print(cv_dataframe)
```

_____

```
    adirlar bahorda  bahorning  erka  go‘zal  guli  kasbini  lola
shifokorlik  tanladi
0        1        1        1        1      1      1      1      0
2        0        0
1        2        0        0        0      0      0      0      1      1
1        1
```

N-gramlar

1. N-gramlar nima va ulardan nima maqsadda foydalaniladi? Quyidagi misollarni ko‘rib chiqamiz [Elov, 2022, 62; Yadav, Borgohain, 2015, 1771]:

1-gap: *“Bu uylar narxi arzon, lekin ular shahar markaziga yaqin emas”*

2-gap: *“Bu mahsulot narxi arzon emas”*

Ushbu misol uchun faqat 8 ta so‘zdan iborat lug‘atni shakllantiramiz:

- uylar
- narxi
- arzon
- shahar
- markaziga
- yaqin
- emas

- mahsulot

Shunday qilib, ushbu gaplar uchun mos vektorlar:

“Bu uylar narxi arzon, lekin ular shahar markaziga yaqin emas” = **[1,1,1,1,1,1,1,0]**

“Bu mahsulot narxi arzon emas” = **[0,1,1,0,0,0,1,1]**

Yuqorida hosil qilingan natijalarda qanday muammo mavjud?

2-gap salbiy, 1-gap esa ijobiy ma’noga ega. Yuqorida shakllantirilgan vektorlarda ushbu ma’lumot aks etmagan. Ushbu muammoni hal qilish uchun N-grammlarni aniqlab olish lozim.

N-gramm – N-tokenli so‘zlar qatoridir: 2 gramm (odatda **bigramm** deb ataladi) ikki so‘zdan iborat *“juda yaxshi”*, *“yaxshi emas”* yoki *“sizning vazifangiz”* va 3-gramm (ko‘pincha **trigramm** deb ataladi) – *“bu mumkin emas”* yoki *“shahar markazidagi uylar”* kabi so‘zlar trigrammni anglatadi.

Masalan, oldingi misoldagi birinchi gapga mos bigrammalar (*“Bu mahsulot narxi arzon emas”*):

- “Bu mahsulot”
- “mahsulot narxi”
- “narxi arzon”
- “arzon emas”

Yuqoridagi misolda oddiy (bitta) so‘zlarni ishlatish o‘rniga, yuqorida ko‘rsatilganidek, **bigrammlardan** (bag-of-bigrams) foydalanamiz. Yaratilgan model asosida 1-gap va 2-gaplar o‘zaro farqlanadi. Shunday qilib, bigrammlardan foydalanish tokenlarni yanada tushunarli ko‘rinishga olib keladi. Xulosa sifatida bigrammlar jamlanmasi so‘zlardan ko‘ra kuchliroqdir va ko‘p hollarda ushbu usuldan foydalanishni tavsiya qilamiz.

Tf-Idf usuli

Yuqorida qo‘llaniladigan baholash usulida har bir so‘zning statistikasi aniqlanib, vektordagi so‘z soni bo‘yicha ifodalandi. Hosil qilingan vektordan hujjatlar haqida ma’lumot olishda foydalanib bo‘lmaydi. Agar biror so‘z hujjatda ko‘p marta uchrasa, bu so‘zning hujjat mazmunida muhimligini yoki aktualligini anglatmaydi [Yadav, Borgohain, 2015; Stecanella, 2019.]

O‘zbek tilidagi juda ko‘p matnlarda “bilan”, “va”, “balki” va shunga o‘xshash so‘zlar juda ko‘p ishlatiladi. Shu sababli ushbu so‘zga mos ball qiymatini kamaytirish lozim. Bu yondashuv term frequency-inverse document frequency yoki qisqacha Tf-Idf deb nomlanadi. TF-IDF usuli yordamida so‘zning berilgan hujjatda qanchalik muhimligi aniqlanadi. Hujjatga mos Tf-Idf qiymat

qanday hisoblanadi? Hujjatdagi so‘z uchun TF-IDF qiymat ikki xil ko‘rsatkichni ko‘paytirish orqali hisoblanadi:

TF-IDF=TF*IDF. Hujjatdagi so‘zning chastotasi (**TF,** term frequency)ni hisoblashning bir necha yo‘li mavjud. Eng oddiyi hujjatda so‘z paydo bo‘lgan holatlarini aniqlash usuli. Shuningdek, chastotani hisoblashning boshqa usullari mavjud. Masalan, so‘z namunalarining dastlabki sonini hujjat uzunligiga yoki hujjatdagi eng ko‘p uchraydigan so‘zning chastotasiga bo‘lish orqali hisoblash mumkin. TF chastotani hisoblash formulasi [Zhu Z., Liang J., Li D., Yu H., Liu G., 2019, 26997.]:

TF (i, j) = n (i, j) / Σn (i, j)

Bu yerda,

- **n (i, j) –** *hujjatda n-chi so‘z necha marta ushrashi;*
- **Σn (i, j) –** *hujjatdagi so‘zlarning umumiy soni*.

Hujjatlar to‘plamidagi so‘zning IDF qiymati hujjatlar to‘plamida so‘zning qanchalik *ko‘p marotaba* yoki *kamdan-kam uchrashi*ni ko‘rsatadi. Qiymat **0** ga qanchalik yaqin bo‘lsa, so‘z shunchalik ko‘p uchraydi. Ushbu ko‘rsatkichni hujjatlarning umumiy sonini aniqlab, ulardagi so‘zni o‘z ichiga olgan hujjatlar soniga bo‘lish va logarifmni hisoblash orqali hisoblash mumkin.

Shunday qilib, agar so‘z hujjatlarda juda ko‘p marotaba ishlatilgan bo‘lsa, IDF qiymat **0** ga, aks holda **1** ga yaqinlashadi. Ushbu ikki qiymatni ko‘paytirish orqali hujjatdagi so‘zning **TF-IDF** qiymatini aniqlash mumkin. Qiymat qanchalik katta bo‘lsa, ushbu hujjatda bu so‘z shunchalik dolzarb (muhim ahamiyatga ega) bo‘ladi. Matematik jihatdan aytganda, TF-IDF qiymat quyidagicha hisoblanadi:

IDF = 1 + log (N/dN)

Bu yerda,

- **N –** *ma’lumotlar to‘plamidagi hujjatlarning umumiy soni;*
- **dN –** *n-o‘rindagi so‘z mavjud hujjatlarning umumiy soni.*

Yuqoridagi formulaga qo‘shilgan 1 qiymat **IDFni silliqlash jarayoni** deb nomlanadi. TF-IDF usulining Python tilidagi tatbig‘ini ko‘rib chiqamiz. Yuqoridagi **BoW (Tf-IDF)** modelini amalga oshirish uchun **Sklearn** kutubxonasidagi **TfidfVectorizer ()** funksiyasidan foydalanish mumkin.

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer,
TfidfVectorizer
```

```
        sentence_1=”Bu uylar narxi arzon, lekin ular shahar
markaziga yaqin emas”
        sentence_2=”Bu mahsulot narxi arzon emas”

        # silliq IDF holida
        print(“Silliq IDF holida:”)
        # tf-idf qiymatni hiosblash
        tf_idf_vec = TfidfVectorizer(use_idf=True,
                        smooth_idf=False,
                        ngram_range=(1,1),stop_
words=[“bu”,”lekin”,”ular”]) # to use only  bigrams ngram_
range=(2,2)
        # transformatsiya
        tf_idf_data = tf_idf_vec.fit_
transform([sentence_1,sentence_2])

        # dataframeni shakllantirish
        tf_idf_dataframe=pd.DataFrame(tf_idf_data.
toarray(),columns=tf_idf_vec.get_feature_names())
        print(tf_idf_dataframe)
        print(“\n”)

        # silliq IDF
        tf_idf_vec_smooth = TfidfVectorizer(use_idf=True,
                        smooth_idf=True,
                        ngram_range=(1,1),stop_
words=[“bu”,”lekin”,”ular”])


        tf_idf_data_smooth = tf_idf_vec_smooth.fit_
transform([sentence_1,sentence_2])

        print(“Silliq IDF:”)
        tf_idf_dataframe_smooth=pd.DataFrame(tf_idf_data_smooth.
toarray(),columns=tf_idf_vec_smooth.get_feature_names())
        print(tf_idf_dataframe_smooth)
        ____________________________

        Silliq IDF holida:
            arzon    emas    mahsulot    markaziga    narxi    shahar
```

uylar yaqin
1 0.262912 0.262912 0.00000 0.445149 0.262912 0.445149 0.445149 0.445149
2 0.412859 0.412859 0.69903 0.000000 0.412859 0.000000 0.000000 0.000000

Silliq IDF:
arzon emas mahsulot markaziga narxi shahar uylar yaqin
1 0.302873 0.302873 0.000000 0.425677 0.302873 0.425677 0.425677 0.425677
2 0.448321 0.448321 0.630099 0.000000 0.448321 0.000000 0.000000 0.000000

BoW usulidagi cheklovlar

BoW usulini amalga oshirish oson bo‘lsa-da, ba’zi kamchiliklar mavjud:

– modelda so‘zning gapdagi joylashuvi inobatga olinmaydi. Misol uchun, “mahsulot arzon”va “mahsulot arzonmi” so‘z birikmalariga mos BoW modelida aynan bir xil vektor hosil qilinadi.

– BoW so‘zning semantikasini inobatga olmaydi. Masalan, “kurash” va “dzyudo” so‘zlari ko‘pincha bir xil kontekstda ishlatiladi. Biroq, bu so‘zlarga mos keladigan vektorlardagi so‘zlar modelida butunlay boshqacha. Gaplarni modellashtirishda bu muammo yanada jiddiylashadi. Masalan: “Ishlatilgan mashinalarni sotib oling” va “Eski mashinalarni sotib oling” BoW modelida mutlaqo boshqa vektorlar bilan ifodalanadi.

– lug‘at hajmi – BoW modeli oldida turgan katta muammo. Misol uchun, agar model hali ishlatilmagan yangi so‘zga duch kelsa, Biblioklept (kitob o‘g‘irlagan degan ma’noni anglatadi) kabi unikal, ammo ma’lumot beruvchi so‘zni aytamiz. BoW modelida bu so‘z inobatga olinmaydi.

Xulosa

BoW – hujjatlarni bir butun sifatida tasniflashning oddiy va qulay usuli hisoblanadi. BoW modeli orqali mashinali o‘rganish algoritmlariga ma’lumotlarni raqamli ko‘rinishda taqdim etish uchun matn modellashtiriladi. BoW modelning aniqligini oshirishning bir necha yo‘llari mavjud:

– BoW modelida unigrammlarni (so‘zlar) ishlatish o‘rniga, bigramm yoki trigrammdan foydalanish model

samaradorligini oshirishga xizmat qiladi.

– TF-IDF modelidan matndagi muhim va nomuhim so‘zlarni aniqlashda foydalaniladi. Odatda TF-IDF modelidan mashinali o‘rganishga asoslangan NLP vazifalarda qo‘llaniladi.

Nisbatan oddiy model bo‘lishiga qaramay, BoW ko‘pincha matn tasnifi NLP vazifalari uchun ishlatiladi.

Adabiyotlar:

Rudkowsky, E., Haselmayer, M., Wastian, M., Jenny, M., Emrich, Š., & Sedlmair, M. More than Bags of Words: Sentiment Analysis with Word Embeddings. Communication Methods and Measures, 12(2–3), 2018. https://doi.org/10.1080/19312458.2018.1455817

Zhang, Y., Jin, R., & Zhou, Z. H. Understanding bag-of-words model: A statistical framework. International Journal of Machine Learning and Cybernetics, *1*(1–4), 2010. https://doi.org/10.1007/s13042-010-0001-0

Yan, D., Li, K., Gu, S., & Yang, L. Network-Based Bag-of-Words Model for Text Classification. IEEE Access, 8, 2020. https://doi.org/10.1109/ACCESS.2020.2991074

Qiu, D., Jiang, H., & Chen, S. Fuzzy information retrieval based on continuous bag-of-words model. Symmetry, 12(2), 2020. https://doi.org/10.3390/sym12020225

Elov, B.B., Hamraeva, Sh., Axmedova, X., Methods for creating a morphological analyze // 14th International Conference on Intellegent Human Computer Interaction. 19-23 October 2022, Tashkent.

Elov, B.B. N-gramm til modellari vositasida o‘zbek tilida matn generatsiya qilish // Kompyuter lingvistikasi: muammolar, yechim va istiqbollar / Xalqaro ilmiy-amaliy konferensiya to‘plami. Elektron nashr / ebook. – Toshkent, 2022.

Yadav, A. K., & Borgohain, S. K. (2015). Sentence generation from a bag of words using N-gram model. Proceedings of 2014 IEEE International Conference on Advanced Communication, Control and Computing Technologies, ICACCCT 2014. https://doi.org/10.1109/ICACCCT.2014.7019414

Stecanella, B. What is TF IDF? MonkeyLearn. 2019.

Zhu, Z., Liang, J., Li, D., Yu, H., & Liu, G. Hot Topic Detection Based on a Refined TF-IDF Algorithm. IEEE Access, 7. 2019. https://doi.org/10.1109/ACCESS.2019.2893980

Pietro, M. di. Text Classification with NLP: Tf-Idf vs Word2Vec vs BERT. 2020. Medium.

Using the bag of words algorithm in natural language processing

Botir Elov[1]
Nizomaddin Xudayberganov[2]
Zilola Xusainova[3]

Abstract:

A bag-of-words model is a digital representation of text to be processed by machine learning algorithms. Using the Bag Of Words (BoW) modeling algorithm, text can be converted and processed into digital matrices. Bag of Words (BoW) is an algorithm that calculates the statistics of a word in a document. The BoW algorithm is used in NLP applications such as document comparison, information retrieval in search engines, document classification, and thematic modeling. This article presents the methods of converting Uzbek texts into digital form using the BoW algorithm.

Keywords: *BoW, Bag of words, set of words, word vector, token, BoW algorithm, TF-IDF method.*

References:

Rudkowsky, E., Haselmayer, M., Wastian, M., Jenny, M., Emrich, Š., & Sedlmair, M. (2018). More than Bags of Words: Sentiment Analysis with Word Embeddings. Communication Methods and Measures, 12(2–3). https://doi.org/10.1080/19312458.2018.1455817

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: A statistical framework. International Journal of Machine Learning and Cybernetics, *1*(1–4). https://doi.org/10.1007/s13042-010-0001-0

[1]*Elov Botir Boltayevich* – doctor of philosophy in technical sciences (PhD), associate professor. Head of the Department of Computer Linguistics and Digital Technologies of Tashkent State University of Uzbek Language and Literature named after Alisher Navoi.
E-mail: elov@navoiy-uni.uz
ORCID: 0000-0001-5032-6648

[1]*Xudayberganov Nizomaddin Uktamboy o'g'li* – Teacher of the Department of Computer Linguistics and Digital Technologies of Tashkent State University of Uzbek Language and Literature named after Alisher Navoi.
E-mail: nizomaddin@navoiy-uni.uz
ORCID: 0000-0002-6213-3015

[2]*Xusainova Zilola Yuldashevna* -Teacher of the Department of Computer Linguistics and Digital Technologies of Tashkent State University of Uzbek Language and Literature named after Alisher Navoi.
E-mail: xusainovazilola@navoiy-uni.uz
ORCID: 0000-0003-4357-7515

For reference: Elov B., Xudayberganov, N., Husainova, Z. 2022. "Using the bag of words algorithm in natural language processing". *Uzbekistan: language and Culture. Applied philology*. 2 (5): 35-50.

Yan, D., Li, K., Gu, S., & Yang, L. (2020). Network-Based Bag-of-Words Model for Text Classification. IEEE Access, 8. https://doi.org/10.1109/ACCESS.2020.2991074

Qiu, D., Jiang, H., & Chen, S. (2020). Fuzzy information retrieval based on continuous bag-of-words model. Symmetry, 12(2). https://doi.org/10.3390/sym12020225

Elov, B.B., Hamraeva, Sh., Axmedova, X., Methods for creating a morphological analyze // 14th International Conference on Intellegent Human Computer Interaction. 19-23 October 2022, Tashkent.

Elov, B.B. N-gramm til modellari vositasida o‘zbek tilida matn generatsiya qilish // Kompyuter lingvistikasi: muammolar, yechim va istiqbollar / Xalqaro ilmiy-amaliy konferensiya to‘plami. Elektron nashr / ebook. – Toshkent: 2022.

Yadav, A. K., & Borgohain, S. K. (2015). Sentence generation from a bag of words using N-gram model. Proceedings of 2014 IEEE International Conference on Advanced Communication, Control and Computing Technologies, ICACCCT 2014. https://doi.org/10.1109/ICACCCT.2014.7019414

Stecanella, B. (2019). What is TF IDF? MonkeyLearn.

Zhu, Z., Liang, J., Li, D., Yu, H., & Liu, G. (2019). Hot Topic Detection Based on a Refined TF-IDF Algorithm. IEEE Access, 7. https://doi.org/10.1109/ACCESS.2019.2893980

Pietro, M. di. (2020). Text Classification with NLP: Tf-Idf vs Word2Vec vs BERT. Medium.