

O'ZBEK MATNLARI UCHUN COUNT VECTORIZER IMKONIYATLARI VA KAMCHILIKLARI

Mahmudjonova Gulshahnoz Ulug'bek qizi

gulshahnozmahmudjonova@gmail.com

ToshDO'TAU 2-kurs magistranti

Annotatsiya. Ushbu maqola matn ma'lumotlarini raqamli tasvirlarga aylantirish uchun tabiiy tilni qayta ishlash (NLP)ning asosiy vositasi bo'lgan CountVectorizer haqida umumiy ma'lumot beradi. U mashinani o'rganish vazifalarida qo'llanilishi, soddaligi va samaradorligi kabi afzalliklari, shuningdek, semantik tushunish kabi kamchiliklarni muhokama qiladi.

Abstract. This article provides an overview of CountVectorizer, a key tool in natural language processing (NLP) for converting text data into text representations. It discusses its application in machine learning tasks, advantages such as simplicity and efficiency, as well as limitations such as the lack of semantic understanding.

Аннотация. В данной статье представлен обзор CountVectorizer, ключевого инструмента в обработке естественного языка (Natural Language Processing, NLP), который используется для преобразования текстовых данных в числовые представления. Обсуждаются его применение в задачах машинного обучения, преимущества такие как простота и эффективность, а также ограничения, такие как отсутствие семантического понимания.

Kalit so'zlar. *Matn, hujjat, normallashtirish, scikit-learn, indeks, vektorlar, NLP.*

NLP (natural language processing – tabiiy tilga ishlov berish) – informatikaning inson tilini tahlil qilish, modellashtirish va tushunish usullari bilan shug'ullanadigan sohasi. Tabiiy til (inson tili) ni o'z ichiga olgan har bir intellektual dasturning asosini NLP tashkil etadi. [Elov, Hamroyeva, Xusainova, 2022:20] Insonda rivojlangan har qanday til tabiiy til hisoblanadi. Tabiiy tilni nutq, qo'shiq aytish, yuz ifodasi, belgilar va tana imo-ishoralari kabi turli shakllarda ko'rish mumkin. NLP 1950-yillarda sun'iy intellekt va tilshunoslikning kesishmasi sifatida boshlangan bo'lib, komputer dasturlariga inson tilini tushunish va qayta ishlashga yordam beradi. Bu sohaning ko'plab vazifalari uchun muhim qadamlardan biri matnlarni, ya'ni so'zlarni raqamlashtirishdir. Bu jarayon matnni oldindan qayta ishlash bo'lib, til korpusida toza bo'lmagan (shovqin, turli xil belgilar, URL

manzillarni o'z ichiga olgan) matnni qayta ishlab dastur o'qiy oladigan va ustida turli vazifalarni amalga oshira oladigan holatga olib kelishdir. Til korpusi matniga dastlabki ishlov berish bosqichlari haqida [Elov, Hamroeva, Alayev, Xusainova, Yodgorov, 2023:144] maqolada to'liq ma'lumot olish mumkin.

Count Vectorizer – bu matn ma'lumotlarini mashinani o'rganish algoritmlari tushunadigan formatga aylantirishga yordam beradigan an'anaviy usul. [<https://towardsdatascience.com/basics-of-countvectorizer>]. CountVectorizer – matnli hujjat to'plamini raqamli ifodaga aylantirish uchun NLP vazifalarida keng qo'llaniladi va matn ma'lumotlarini tokenizatsiya qilish va har bir tokenning paydo bo'lishini hisoblash orqali ishlaydi [Neri, 2023]. CountVectorizer – matnli hujjatlar to'plamini raqamli tasvirga aylantirish uchun tabiiy tilni qayta ishlashda keng qo'llanilib, matnni tasniflash, klasterlash va ma'lumotlarni qidirishni o'z ichiga olgan ko'plab NLP vazifalarini qayta ishlash bosqichida asosiy vositadir. CountVectorizer dasturining asosiy g'oyasi hujjat atamasi matritsasini yaratishdan iborat bo'lib, unda har bir satr matn korpusidagi hujjatni, har bir ustun esa korpusda topilgan unikal so'z yoki “termin”ni ifodalaydi. Matritsadaqi qiymatlar har bir hujjatdagi har bir atamaning paydo bo'lish chastotasini ko'rsatadi.

CountVectorizer afzalliklari

Oddiy: CountVectorizerdan foydalanish va tushunish oson. U o'ziga xos parametrlarga ega va matnni oldindan qayta ishlashni boshlash uchun kichik konfiguratsiyani talab qiladi.

Tez va samarador: hisoblash jihatidan samarali va ko'plab hujjatlar bilan katta matnli ma'lumotlar to'plamini boshqara oladi. U kompyuter xotirasini tejaydi va qisqa vaqtda natija qaytaradi, ayniqsa yuqori o'lchamli ma'lumotlar bilan ishlashda. Shuningdek, u nomuhim so'zlarini filtrlaydi va lug'at hajmini nazorat qilish imkoniyatlarini beradi.

CountVectorayzerning kamchiliklari

Semantik ma'noni e'tiborsiz qoldiradi: U har bir tokenga alohida obyekt sifatida qaraydi va so'zlar orasidagi semantik munosabatlarni saqlamaydi. U kontekstdagi so'zlar ma'nosini hisobga olmaydi, bu esa so'z semantikasini tushunishni talab qiladigan vazifalarda uning samaradorligini cheklashi mumkin.

Hujjat uzunligini hisobga olmaydi: hajm jihatdan katta bo'lgan hujjatlar qisqaroq hujjatlarga qaraganda ko'proq belgilar soniga ega bo'lsa ham ma'no tarafdin ulardan farq qilishi mumkin. Ushbu normalizatsiya yetishmasligi hujjat uzunligiga tayanadigan maxsus tahlillar va algoritmlarga ta'sir qilishi mumkin.

Cheklangan ma'lumot: faqat hujjatlardagi tokenlarning chastotasini ushlaydi. U his-tuyg'ularni tahlil qilish yoki tilni modellashtirish kabi maxsus matn tahlili vazifalarida tegishli bo'lishi mumkin bo'lgan so'zlarning tartibi yoki ketma-ketligini hisobga olmaydi.

Xato xulosa: Yuqori chastotali so'zlar matn uchun muhimroq ahamiyatga ega ma'lumot beradi degan xato xulosani beradi. “bilan, va, ammo..” kabi nomuhim so'zlarni bunga misol sifatida keltirish mumkin [Elov, Hamroyeva, Xusainova, Xudayberganov, 2023:81].

CountVectorizerdan olingan hujjat muddatli matritsasi talqin qilinadigan natijalarni beradi. Matritsada har bir katak ma'lum bir hujjatdagi tokenning sonini yoki chastotasini ifodalaydi, bu to'g'ridan-to'g'ri tahlil qilish va o'rganish imkonini beradi. CountVectorizer, Pythonning scikit-learn kutubxonasida joylashgan. Ushbu kutubxona ma'lumotlar bilan ishlash, o'rganish, va ma'lumotlar analizi uchun ishlatiladi. CountVectorizer esa matnlar bilan ishlashda o'rtacha bo'lgan bir vositadir, u matn ma'lumotlarini raqamlar bilan ifodalaydi. Batafsilroq tushunish uchun, quyidagi matnlar asosida tahlil qilinadi.

CountVectorizer – bu Pythondagi scikit-learn kutubxonasi tomonidan taqdim etilgan. U berilgan matnni butun hujjatda paydo bo'ladigan har bir so'zning chastotasi (soni) asosida vektorga aylantirish uchun ishlatiladi. Bu bir nechta matnlar mavjud bo'lganda va har bir matndagi har bir so'zni vektorlarga aylantirmoqchi bo'linganda Count Vectorizerdan foydali bo'ladi (matn tahlilida foydalanish uchun). Quyida hujjatdagi bir nechta matn misollari orqali ko'rib chiqiladi (har biri (list) ro'yxat elementi sifatida):

hujjat = [“Men dasturlashni yaxshi bilaman.”, “Dasturlash doimiy bilimlar olishni talab qiladi.”, “Men kod yozishni yoqtiraman.”] quyidagi jadvalda hujjat to'plami raqamli ifodaga (matritsa) aylantirilgan bo'lib, ustunlar hujjatlarni, qatorlar esa unikal so'zlarni ifodalovchi matritsa hosil qilgan (2-jadval).

Unikal so'zlar	Hujjat [0]	Hujjat [1]	Hujjat [2]
bilaman	1	0	0
dasturlashni	1	0	0
olishni	0	1	0
men	1	0	1
dasturlash	0	1	0
kod	0	0	1
qiladi	0	1	0
yaxshi	1	0	0
yoqtiraman	0	0	1

bilimlar	0	1	0
doimiy	0	1	0
yozishni	0	0	1
talab	0	1	0

1-jadval. Unikal soʻzlar

Hujjatda jadval qatorlari sifatida ifodalangan 13 ta unikal soʻz mavjud.

1. Hujjatda 3 ta matn namunasi mavjud boʻlib, ularning har biri jadval ustunlari sifatida koʻrsatilgan.
2. Har bir katak oʻsha matndagi soʻzning sonini koʻrsatadigan raqamni oʻz ichiga oladi.
3. Barcha soʻzlar kichik harflarga oʻtkaziladi.

CountVectorizer ichida bu soʻzlar satr, yaʼni soʻz sifatida saqlanmaydi. Aksincha, ularga maʼlum bir indeks qiymati beriladi. Bunday holda, “bilaman” indeksi 0, “dasturlashni” indeksi 1, “men” indeksi 3 kabi ifodalanadi. Unikal soʻzlar indeksi quyidagi jadvalda koʻrsatilgan (2-jadval).

Indeks	Hujjat [0]	Hujjat [1]	Hujjat [2]
0	1	0	0
1	1	0	0
2	0	1	0
3	1	0	1
4	0	1	0
5	0	0	1
6	0	1	0
7	1	0	0
8	0	0	1
9	0	1	0
10	0	1	0
11	0	0	1
12	0	1	0

2-jadval. Unikal soʻzlar indeksi

Python dasturidan foydalanib CountVectorizer ishlatiladi:

```
from sklearn.feature_extraction.text import CountVectorizer
hujjat = [ "Men dasturlashni yaxshi bilaman.",
           "Dasturlash doimiy bilimlar olishni talab qiladi.",
```

```
“Men kod yozishni yoqtiraman.”
]
# CountVectorizer obyektini yaratish
vectorizer = CountVectorizer ()
vectorizer.fit(document)
# Belgilangan noyob soʻzlarni indeksleri bilan birga chop etish
print (“Vocabulary:”, vectorizer.vocabulary_)
# Hujjatni kodlashtirish
vector = vectorizer.transform(document)
# Kodlangan matnlarni umumlashtirish
print (“Encoded Document is:”)
print (vector.toarray())
#Natija
Lugʻat: {'men': 6, 'dasturlashni': 3, 'yaxshi': 10, 'bilaman': 0, 'dasturlash': 2, 'doimiy': 4,
'bilimlar': 1, 'olishni': 7, 'talab': 9, 'qiladi': 8, 'kod': 5, 'yozishni': 12, 'yoqtiraman': 11}
Kodlangan hujjat:
[[1 0 0 1 0 0 1 0 0 0 1 0 0]
 [0 1 1 0 1 0 0 1 1 1 0 0 0]
 [0 0 0 0 0 1 1 0 0 0 0 1 1]]
```

Count Vectorizer oʻziga xos afzallik va kamchiliklarga ega. Undan foydalanishda matn ustida bajariladigan amalni hisobga olish kerak. Maqolada Count Vectorizerdan sodda dastur uchun sinab koʻrildi. Keyingi izlanishlarda ixtiyoriy matn tanlanib, unda soʻzlar statistikasini aniqlash kabi amaliyotlar bajariladi.

Foydalanilgan adabiyotlar:

1. Elov B., Hamroyeva Sh., Xusainova Z., “NLP (tabiiy tilga ishlov berish)ning vazifalari va zamonaviy yondashuvlar” Filologik tadqiqotlar: til, adabiyot, taʼlim, 2022. - B. 20
2. Elov, B., Hamroyeva, Sh., Alaev, R., Xusainova, Z., Yodgorov, U., (2023). Methods of processing the Uzbek language corpus texts.
3. <https://towardsdatascience.com/basics-of-countvectorizer>
4. Neri Van Otten., “CountVectorizer Tutorial In Scikit-Learn And Python (NLP) With Advantages, Disadvantages & Alternatives” <https://spotintelligence.com/2023/05/17/countvectorizer/> 2023
5. Elov B, Hamroyeva Sh, Xusainova Z, Xudayberganov N (2023). “Oʻzbek tili korpusi matnlarini qayta ishlashda CountVectorizer, TF-IDF hamda Co-occurrence matrix usullarining ahamiyati” Elektron lugʻatlar yaratishning nazariy va amaliy asoslari mavzusidagi xalqaro ilmiy-amaliy anjuman materiallari ., Andijon-2023. – B. 81