LEKSIK SINONIMLARNI ANIQLASH UCHUN WORD2VEC VA ROBERTAFORMASKEDLM MODELLARIDAN FOYDALANISH. SYN-ROBERTA MODELI HAQIDA

Uzoqova Mohiyaxon Tuyg‘un qizi
ToshDO‘TAU
Kompyuter lingvistikasi mutaxassisligi 1-kurs magistranti
mohiyaxonuzokova@gmail.com

Annotatsiya. Leksik sinonimlarni lug‘atga asoslangan usul, ya’ni avvaldan shakllantirilgan sinonimlar bazasi bilangina avtomatik aniqlash to‘laqonli o‘zini oqlamasligi mumkin. Sababi sinsonim sifatida taqdim etilgan so‘zlar kontekstga mos tushmasligi mumkin. Bu esa o‘zbekcha sinonimayzer dasturidan foydalanishda noqulaylik tug‘diradi. Ushbu maqolada sinonimlarni kontekstni inobatga olgan holda aniqlashda foydali deb topilayotgan Word2Vec va RoBERTa modellariga nazar tashlandi. Qolaversa, Word2Vec, RoBERTa modeli va barcha uslub va ohang doirasida teglangan sinsetlar bazasi integratsiyasi asosida ishlaydigan Uz-Synonymizer loyihasiga to‘xtalib o‘tildi.

Kalit so‘zlar: Word2Vec, RoBERTa, vektorayzer, tokenayzer, mashinali o‘qitish, leksik sinonimlar.

Abstract. A dictionary-based method - automatic identification of lexical synonyms with a previously formed synonym database may not fully justify itself. This is because words presented as synonyms may not fit the context. This makes it difficult to use the Uzbek synonymizer program. This article looks at the Word2Vec and RoBERTa models that have been found to be useful in context-aware synonym detection. In addition, information was provided about the Uz-Synonymizer project, which works on the integration of Word2Vec, the RoBERTa model, and a database of tagged synsets within all styles and tones.

Key words: Word2Vec, RoBERTa, vectorizer, tokenizer, machine learning, lexical synonyms.

Аннотация. Словарный метод - автоматическая идентификация лексических синонимов с заранее сформированной базой синонимов может не полностью себя оправдать. Это связано с тем, что слова, представленные как синонимы, могут не соответствовать контексту. Это затрудняет использование программы узбекского синонимайзера. В этой статье рассматриваются модели Word2Vec и RoBERTa, которые оказались полезными для обнаружения синонимов с учетом контекста. Кроме того, была предоставлена информация о проекте Uz-Synonymizer, работающем над интеграцией Word2Vec, модели RoBERTa и базы данных тегированных синсетов во всех стилях и тональностях.

Ключевые слова: Word2Vec, RoBERTa, векторизатор, токенизатор, машинное обучение, лексические синонимы.

Foydalanuvchi tomonidan kiritilgan so‘rovga (so‘z yoki matn) muvofiq so‘z [1], so‘z birikmasi, ba’zan gaplarga [2] sinonim taklif qiladigan yoki avtomatik ravishda sinonimlarga almashtiruvchi dastur – o‘zbekcha sinonimayzerni qurishda, avvalo, leksik sinonimlarni avtomatik aniqlash masalasini hal etish dolzarbdir. Leksik sinonimlashni amalga oshirish uchun **lug‘atga asoslangan usul** bilan cheklanish o‘zini oqlamasligini, chunki, avvalo, muqobil javoblar kontekstga mos tushmasligi yoki semantik konnotatsiya e’tibordan chetda qolishi mumkinligini ilgarigi tadqiqotlarimizda ham ta’kidlagan edik [Uzoqova, 2023: 288]. Xususan, sinonimayzerni yaratishda sinonimik korpusni yaratishning o‘zi yetarli bo‘lmasligi mumkin. Bunda semantik maydon, semantik freym, uyadoshlik, polifunksionallik, sentiment kabi inobatga olinishi lozim bo‘lgan til hodisalari mavjud. Demak, **mashinali o‘qitish usuliga** murojaat qilish kerak. Xo‘sh, qanday qilib mashina yordamida bunday muammolarni hal qilish mumkin? Namunali “til modeli”ni shakllantirish va uni kompyuterga integratsiya qilishning qanday yo‘llari bor?

Jahon tajribasida semantik o‘rindoshlik va yaqin ma’noli leksemalarni aniqlash bo‘yicha salmoqli ishlar amalga oshirilmoqda. Jumladan, Word2Vec va BERTga asoslangan model [4] kontekstni inobatga olgan holda sinsetlardan unumli foydalanish imkonini beradi.

Word2Vec modeli.

Word2Vec – matndagi har bir unikal tokenni vektorli ko‘rinishga o‘tkazadigan va bu vektorlarning bir-biriga munosabatiga bog‘liq bo‘lgan, neyron tarmoqlar bilan ishlaydigan model. Modelning mohiyatiga ko‘ra, vektorlar fazosida kosinus masofasi 1 ga yaqin bo‘lgan vektorlar o‘xshash ma’noli deb qaraladi [5].

Word2vec modeli so‘zlarning sinonimlarini aniqlashda [6], mashina tarjimasida, avtomatik teglashda [5] ishlatiladi.

O‘zbekcha matnlarda leksik sinonimlarni Word2Vec modeli yordamida aniqlash uchun biz quyidagi ketma-ketlik asosida ish olib bordik:

I bosqich. Vektorlar fazosini hosil qilish.

1-qadam. Modelni mashq qildirish uchun ixtiyoriy matn shakllantirildi:

Men ilgaridan go‘zal rasmlar shaydosiman. Yaqinda ajoyib rasmlar ko‘rgazmasiga bordim. U yerda chiroyli rasmlar ko‘p edi. Ayniqsa, peyzaj va natyurmort janridagi rasmlar go‘zal yozilgan edi.

2-qadam. Bu matndan unikal so‘zlar ro‘yxati hosil qilindi:

- *Ajoyib;*
- *ayniqsa;*
- *bordim;*
- *chiroyli;*
- *natyurmort;*
- *peyzaj;*
- *rasmlar;*
- *shaydosiman;*

- *edi;*
- *go‘zal;*
- *ilgaridan;*
- *janridagi;*
- *ko‘p;*
- *ko‘rgazmasiga;*
- *men;*
- *u;*
- *va;*
- *yaqinda;*
- *yerda;*
- *yozilgan*

3-qadam. Hosil bo‘lgan har bir unikal so‘z uchun ko‘rib chiqiladigan kontekstli so‘zlar yoki tokenlar soni (“window size”) belgilab olindi va ularning o‘zaro birga kelishlari soni (chastotasi) hisoblandi (1-rasm). Masalan, biz mashq uchun tanlab olgan matnimizda hisoblash jarayoni quyidagicha amalga oshirildi (1-rasm). Garchi Word2Vec modelining sozlamalarida kontekstli o‘lchov soni standart holatda 5 bo‘lsa-da, matnning hajmini inobatga olgan holda biz o‘lchamni 2 ga tushirishga qaror qildik (window size =2).

Unikal so‘zlar	Ajoyib	Bordim	Chiroyli	Edi	Go‘zal	Ilgaridan	Ko‘p	Ko‘rgazmasiga	Men	Rasmlar	Shaydosiman	U	Yerda	Yaqinda	Ayniqsa	Peyzaj	Va	Natyurmort	Janridagi	Yozilgan
Ajoyib	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
Ayniqsa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
Bordim	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
Chiroyli	0	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	0
Edi	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1
Go‘zal	0	0	0	1	0	1	0	0	1	2	1	0	0	0	0	0	0	0	1	1
Ilgaridan	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
Janridagi	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0
Ko‘p	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Ko‘rgazmasiga	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Men	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Natyurmort	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1	0
Peyzaj	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0
Rasmlar	1	1	1	1	2	1	1	1	0	0	1	0	1	1	0	0	0	1	1	1
Shaydosiman	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
U	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Va	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0
Yaqinda	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Yerda	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
Yozilgan	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

1-rasm. Unikal so‘zlarning vektorlarga o‘tkazilishi

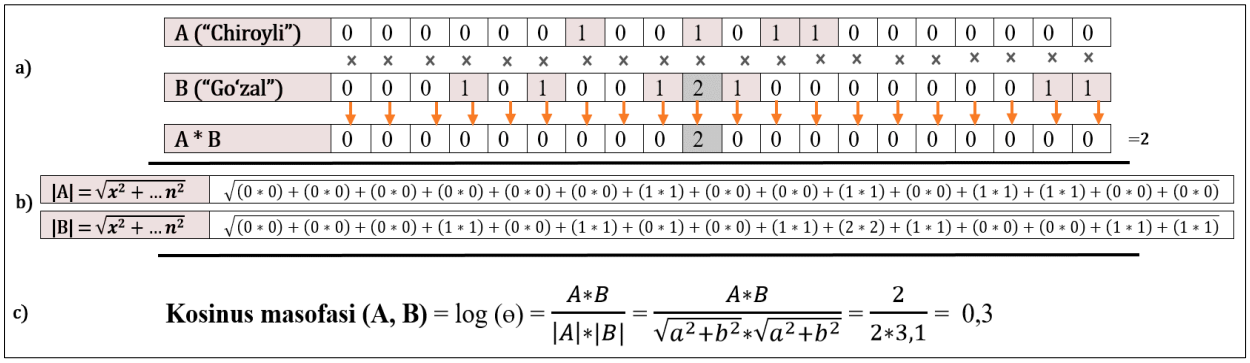
Hosil bo‘lgan har bir gorizontal qator unikal so‘zga tegishli vektor sifatida baholanadi. Hosil bo‘lgan vektorlar so‘zlar juftligi bilan modelda saqlanadi.

II bosqich. Vektorlar orqali sinonimlarni topish.

1-qadam. Matnda uchragan ixtiyoriy so‘z (bizning holatimizda “chiroyli” so‘zi) modelga sinov uchun berildi. Model esa bu so‘zning vektorini aniqlab oldi:

A(“Chiroyli”) = [0,0,0,0,0,0,1,0,0,1,0,1,1,0,0,0,0,0,0,0]

2-qadam. Tanlangan so‘zning vektori bilan qolgan barcha so‘zlarning vektorlari kosinus masofalari aniqlandi (2-rasm). Namunada “chiroyli” so‘zining “go’zal” so‘zi bilan kosinus masofasi hisoblangan.



2-rasm. “Chiroyli” va “go‘zal” so’zlari vektorlari orasidagi kosinus masofani hisoblash

a) A(“chiroyli”) va B(“go‘zal”) vektorlaring skalyar ko‘paytmasi hisoblandi;

b) A va B vektorlarning uzunliklari hisoblandi;

c) A va B vektorlarning skalyar ko‘paytmasi ularning uzunliklariga bo‘linib ikkala vektor orasidagi kosinus masofa hisoblandi.

3-qadam. Hisoblangan so‘zlar va ularning kosinus masofalari kamayish tartibida qaytarildi (3-rasm).

```
>>> from gensim.models import Word2Vec
>>> model = Word2Vec.load("models/word2vec/test.model")
>>> print(*model.wv.most_similar('chiroyli', topn=5), sep = '\n')
('bordim', 0.2528955936431885)
('natyurmort', 0.14256243407726288)
('go‘zal', 0.13727802038192755)
('va', 0.11664504557847977)
('ilgaridan', 0.04409514367580414)
>>>
```

3-rasm. “Chiroyli” so‘zining sinonimlari va ular orasidagi kosinus masofalar

Natijadan ko‘rinib turibdiki, kontekstli so‘zlar o‘lchovi asosida chiroyli so‘zining vektoriga kosinus masofasi eng yaqin vektorlar yuqoridagi 5 ta so‘zdir. Ammo tanlangan so‘zga (“chiroyli”) sinonim sifatida qaytarilgan so‘zlar (“bordim”, “natyurmort”, “go‘zal”, “va”, “ilgaridan”) aslida *sinonim deb atashga yaroqli emas*. Buning asosiy sababi mashq matni hajmining kichkinaligidir. Nisbatan ishonchli natijalarni olish uchun biz mashq matni hajmini kengaytirishga qaror qildik. Buning uchun daryo.uz, kun.uz kabi davriy nashrlar asosida matnlarni avtomatik yig‘ib oldik, yagona hujjatga aylantirdik [7] va mashq uchun modelga taqdim etdik (4-rasm).

```
>>> from gensim.models import Word2Vec
>>> model = Word2Vec.load("models/word2vec/word2vec.model")
>>> print(*model.wv.most_similar('chiroyli', topn=5), sep = '\n')
('jozibali', 0.7959373593330383)
('ajoyib', 0.7799746990203857)
('go‘zal', 0.7486552000045776)
('maftunkor', 0.738379716873169)
('didsiz', 0.7264962196350098)
>>>
```

4-rasm. “Chiroyli” so‘zining yangi mashq-matn asosida topilgan sinonimlari va ular orasidagi kosinus masofalar

To‘plangan matnning hajmi yirikligi bois “chiroyli” so‘ziga kosinus masofasi eng yaqin kelgan dastlabki to‘rtta so‘z kontekstga ko‘ra haqiqatdan ham sinonim sifatida xizmat qilishi mumkin. Ammo beshinchi so‘z (“didsiz”) modelning aynan *sinonimlarni aniqlay olish vazifasiga qay darajada ishonish mumkin*, degan savolni oldimizga ko‘ndalang qo‘yadi. Ya’ni Word2Vec modeli orqali olinadigan natijalar doim ham berilgan so‘zning real sinonimi bo‘la olmaydi. Aksincha, kontekstli so‘zlar o‘lchamiga qarab antonimlarni ham sinonim qatorida taklif qilishi mumkin. Demak, shartli ravishda xulosa qilish mumkinki, Word2Vec modeli statistik tahlilga ko‘ra o‘xshash (to‘liq sinonim emas) vektorlarni qaytaruvchi model. Shu sababli, bizningcha, leksik sinonimlari mashinali o‘qitish modellari yordamida aniqlaganda Word2Vecning o‘zi bilan cheklanib qolmagan ma’qul.

RoBERTaForMaskedLM modeli.

RoBERTa (Robustly Optimized BERT Pretraining Approach – BERT modelini ilgaridan o‘qitishga asoslangan puxta optimallashtirilgan model) ForMaskedLM modeli BERT modelining [Devlin va b, 2019: 4171] kuchaytirilgan varianti bo‘lib [Liu va b., 2019], neyron tarmoq bilan ishlovchi zamonaviy model hisoblanadi. Bu model 2019-yilda FacebookAI guruhi tomonidan ishlab chiqilgan

[10] va turli maqsadlarda: mashina tarjimasida, tabiiy tilni qayta tushunishda va matn tasnifida ishlatilishi mumkin [10].

Modelning asosiy vazifasi matndan maqsadli tushirib qoldirilgan tokenga (“masked”) mos tokenlarni taqdim etishdir. Biz modelning ushbu vazifasidan kelib chiqqan holda uni matndagi leksik sinonimlarni topish uchun yo‘naltirishga qaror qildik. Ya’ni *RoBERTa modeli bilan gapdagi bir so‘zning sinonimlari shu gapga (kontekstga) mos kelishini tekshirdik*. Jarayon quyidagi ketma-ketlikda amalga oshirildi.

I bosqich. RoBERTa uchun tokenayzerni moslashtirish.

1-qadam. HuggingFace jamiyati tomonidan ishlab chiqilgan *tokenizers* kutubxonasi orqali avvaldan mashq qildirilgan tokenayzer chaqirildi (import qilindi) va berilgan matn uchun sozlamalari moslandi (5-rasm).

```
12  from tokenizers import ByteLevelBPETokenizer
13  dataset_file = 'data/test.txt'
14  tokenizer = ByteLevelBPETokenizer()
15
16  tokenizer.train(
17      files=dataset_file,
18      vocab_size=52_000,
19      min_frequency=1,
20      special_tokens=["<s>", "<pad>", "</s>", "<unk>", "<mask>",])
```

5-rasm. Tokenayzerni chaqirish va berilgan matn uchun sozlamarni moslash

Berilgan matn: *Men ilgaridan go’zal rasmlar shaydosiman. Yaqinda ajoyib rasmlar ko’rgazmasiga bordim. U yerda chiroyli rasmlar ko’p edi. Ayniqsa, peyzaj va natyurmort janridagi rasmlar go‘zal yozilgan edi.*

2-qadam. Tokenayzer matndan (datasetdan) o‘zi uchun unikal tokenlarni yasadi va ularga tartib raqami (ID) berdi. Sozlamalarda yasama tokenlarning maksimal hajmi (“vocab_size”, 5-rasm) standart holat uchun 52 000 gacha deb belgilangan. Bizning holatimizda unikal yasama tokenlar miqdori 369 taga yetdi.


```
{"<s>":0,"<pad>":1,"</s>":2,"<unk>":3,"<mask>":4,"!":5,"\"":6,"#":7,"$":8,"%":9,"&":10,"'":11,
"(":12,")":13,"*":14,"+":15,",":16,"-":17,".":18,"/":19,"0":20,"1":21,"2":22,"3":23,"4":24,
"5":25,"6":26,"7":27,"8":28,"9":29,":":30,";":31,"<":32,"=":33,">":34,"?":35,"@":36,"A":37,
.....
"qinda":352,"rgazmas":353,"ridagi":354,"yurmort":355,"Ġilgaridan":356,"ĠAyniqsa":357,
"ĠYaqinda":358,"Ġajoyib":359,"Ġbordim":360,"Ġchiroyli":361,"Ġjanridagi":362,"Ġnatyurmort":363,
"Ġpeyzaj":364,"Ġshaydosim":365,"Ġyerda":366,"Ġyozilgan":367,"rgazmasiga":368,"Ġshaydosiman":369}
```


6-rasm. Berilgan matnning tokenlarga ajratilishi va raqamlanishi

3-qadam. Hosil qilingan tokenlar saqlanadigan jildning manzili ko‘rsatildi. Model ko‘rsatilgan jildda ikkita fayl hosil qilindi: vocab.json va merges.json. Tokenayzerning vazifasi bu jarayonda yakuniga yetdi.

II bosqich. RoBERTa modelini mashq qildirish.

1-qadam. Model uchun sozlamalar yozildi va bu sozlamalar asosida RobertaForMaskedLM modeli hosil qilindi (7-rasm).

```
config = RobertaConfig(
    vocab_size=52_000,
    max_position_embeddings=514,
    num_attention_heads=12,
    num_hidden_layers=6,
    type_vocab_size=1,
)
model = RobertaForMaskedLM(config=config)
```

7-rasm. RoBERTa modeli uchun sozlamalarni moslash

```
dataset = LineByLineTextDataset(
    tokenizer=tokenizer,
    file_path=dataset_file,
    block_size=128,
)
```

8-rasm. Berilgan matnni tokenayzer natijasiga muvofiq raqamlash

2-qadam. Berilgan matn avvaldan hosil qilingan tokenayzer yordamida raqamli ko‘rinishga o‘tkazildi (8-9-rasmlar).

<s>	Men	ilgaridan	go	‘	zal	rasmlar	shaydosiman	.	Yaqinda
0	348	356	283	269	285	270	369	18	358
ajoyib	rasmlar	ko	‘	rgazmasiga	bordim	.	U	yerda	chiroyli
359	270	284	269	368	360	18	328	366	361
rasmlar	ko	‘	p	edi	.	Ayniqsa	,	peyzaj	va
270	284	269	84	282	18	357	16	364	339
natyurmort	janridagi	rasmlar	go	‘	zal	yozilgan	edi	.	</s>
363	362	270	283	269	285	367	282	18	2

9-rasm. Berilgan matnni tokenayzer natijasiga muvofiq raqamlashning jadval ko‘rinishida aks etishi

3-qadam. Modelni mashq qildirish uchun argumentlarni yozildi (10-rasm).

```
training_args = TrainingArguments(
    output_dir="models/UzRoBERTa", overwrite_output_dir=True, num_train_epochs=1,
    per_device_train_batch_size=8, save_total_limit=2, prediction_loss_only=True,
    save_steps=10_000
)
trainer = Trainer(
    model=model, args=training_args, data_collator=data_collator, train_dataset=dataset,
)
```

10-rasm. Modelni mashq qildirish uchun argumentlarni yozish

4-qadam. Model mashq qildirildi va argumentda ko‘rsatilgan joyga saqlandi.

III bosqich. Modeldan foydalanish.

Model saqlangan joydan qayta yuklab olindi va unga test uchun gap berildi. Test sifatida berilayotgan gapdan bir so‘z <*mask*> tokeni ostida maqsadli ravishda tushirib qoldirildi. Namuna uchun *“U yerda chiroyli rasmlar ko‘p edi”* gapi olindi (11-rasm).

```
from transformers import pipeline

fill_mask = pipeline(
    "fill-mask",
    "models/UzRoBERTa"
)
print(*fill_mask("U yerda <mask> rasmlar ko'p edi", top_k = 5), sep='\n')
```

11-rasm. Model yordamida tushib qolgan so‘zni topish

Quyidagi natija olindi:

```
C:\Disc\Projects\uz-synonimizer\test>python tester_roberta.py
{'score': 0.0001581029937369749, 'token': 285, 'token_str': 'zal', 'sequence': 'U yerdazal rasmlar ko’p edi'}
{'score': 0.00014050737081561238, 'token': 7835, 'token_str': '', 'sequence': 'U yerda rasmlar ko’p edi'}
{'score': 0.00013314350508153439, 'token': 43643, 'token_str': '', 'sequence': 'U yerda rasmlar ko’p edi'}
{'score': 0.00012822699500247836, 'token': 26899, 'token_str': '', 'sequence': 'U yerda rasmlar ko’p edi'}
{'score': 0.00012391315249260515, 'token': 13471, 'token_str': '', 'sequence': 'U yerda rasmlar ko’p edi'}
```

12-rasm. Sinov uchun berilgan gap asosida olingan eng yuqori beshta natija

Natijadan ko‘rinib turibdiki, birorta token ma’noga ega emas. Faqatgina birinchi token “go‘zal” so‘zining ikkinchi bo‘g‘inini takrorlagani bois “chiroyli” so‘ziga shartli ravishda yaqin deb qarash mumkin. Modelning bunday natija qaytarishiga asosiy sabab datasetning hajmi kichkinaligidir. Shuning uchun biz Word2Vec modelida sinaganimiz kabi kattaroq dataset [7] bilan natija olishga

```
C:\Disc\Projects\uz-synonimizer>python tester_roberta.py
{'score': 0.018685689195990562, 'token': 580, 'token_str': ' G', 'sequence': "U yerda G rasmlar ko'p edi"}
{'score': 0.01854391023516655, 'token': 497, 'token_str': ' «', 'sequence': "U yerda « rasmlar ko'p edi"}
{'score': 0.01280809286981821, 'token': 360, 'token_str': ' A', 'sequence': "U yerda A rasmlar ko'p edi"}
{'score': 0.009370371699333191, 'token': 665, 'token_str': ' E', 'sequence': "U yerda E rasmlar ko'p edi"}
{'score': 0.008777589537203312, 'token': 433, 'token_str': ' K', 'sequence': "U yerda K rasmlar ko'p edi"}
```

harakat qildik (13-rasm):

13-rasm. Sinov uchun berilgan gap bo‘yicha yangi mashq-matn orqali olingan eng yuqori beshta natija

Biroq bu safar ham “chiroyli” so‘zi o‘rniga *“G”, “,”, “A”, “E”, “K”* kabi ma’nosiz natijalarni oldik. Buning asosiy sababi test sifatida olingan gapdagi so‘zlar butun hujjat davomida juda kam o‘zaro aloqaga kirishgani deb qaraldi. Modelni boshqa namuna bilan sinashda davom etdik. Sinov uchun ***“Bugun Toshkent <mask> anjuman bo‘lib o‘tdi”*** gapini berdik. Eng yuqori 100 ta natijadan sanoqlilarigagina nisbatan yaroqli deb qarash mumkin:

score	token	token_str	sequence
0.06776456534862518	1417	shahrida	Bugun Toshkent **shahrida** anjuman bo'lib o'tdi
0.011984631419181824	774	viloyati	Bugun Toshkent **viloyati** anjuman bo'lib o'tdi
0.0024553691036999226	264	da	Bugun Toshkent**da** anjuman bo'lib o'tdi
0.0002205256314482 54	2191	hududida	Bugun Toshkent **hududida** anjuman bo'lib o'tdi
0.00011574733798624948	501	larda	Bugun Toshkent**larda** anjuman bo'lib o'tdi

RobertaForMaskedML modeli, guvoh bo‘ldikki, datsetning hajmi qancha yirik bo‘lsa, shuncha sifatli ishlaydi. Masalan, **RoBERTa base** modelining o‘zini mashq qildirish uchun jami 160 GB hajmdagi matndan foydalanilgan [10]. Natijalar esa *ishonchli* va *yaroqli*.

Demak, yuqorida ta’kidlangandek, gapdagi muayyan so‘zning sinonimlari shu shu kontekstga mos kelishini tekshira olishimiz uchun bizga yirik hajmdagi sifatli va avvaldan o‘qitilgan dataset kerak [11].

Uz-Synonymizer loyihasi.

Leksik sinonimlarni matnda mashinali o‘qtish algoritmlaridan foydalangan holda topish vazifasini amalga oshirish uchun esa Word2Vec va RobertaForMaskedLM modellarini, bizningcha, avvaldan shakllantirilgan sinsetlar bazasi bilan integratsiya qilish lozim. Biz bu uchta hodisani **Uz-Synonymizer** loyihasi nomi ostida birlashtirdik.

Loyihaning umumiy ishlash prinsipi: berilgan matn gaplarga bo‘linadi. Gapdagi har bir so‘z uchun Word2Vec modeli va sinsetlar bazasi orqali o‘sha so‘zning sinonimlari tanlab olinadi. So‘ng sinonimlari aniqlangan so‘zlar berilgan gapda <*mask*> tokeniga almashtiriladi va RoBERTa modeliga taqdim etiladi. RoBERTa modeli esa o‘z navbatida <*mask*> tokeni o‘rniga sintaktik va semantik jihatdan mos bo‘lgan tokenlarni taqdim etadi. Shu tokenlar seti va oldindan shakllantirilgan sinsetda mavjud tokenlar berilgan gapdagi so‘zlarning kontekstga mos sinonimi deb topiladi.

Masalan,

Berilgan gap	***Uz-Synonymizer loyihasida***
Tadbirkorlik – foyda olish uchun faoliyat	Tadbirkorlik – foyda **qilish** uchun faoliyat.
	0.09138742834329605
	Tadbirkorlik – foyda olish **kabi** faoliyat.
	0.009929370135068893
	Tadbirkorlik – **pul** olish uchun faoliyat.
	0.004994241986423731
	Tadbirkorlik – **manfaat** olish uchun faoliyat.
	0.003256471361964941
	Tadbirkorlik – foyda olish **haqida** faoliyat.
	0.0010812038090080023

Xulosa sifatida shuni aytish mumkinki, sinsetlar bazasi integratsiya qilingan taqdirda ham Uz-Synonymizer loyihasi muvaffaqiyatli ishlashi uchun nihoyatda ulkan hajmdagi toza va sifatli ma’lumotlar bazasini talab qilinadi. Datasetni shakllantirishda informatsion mazmundagi davriy nashrlar bilan cheklanib qolmay, ilmiy, publisistik va adabiy til doirasida badiiy uslubda yozilgan (lirik asarlar bundan mustasno) matnlarni ham qamrab olish lozim. Ana shunda modellar yuqori aniqlikda natija qaytarishi mumkin.

Qolaversa, morfologik, sintaktik va frazeologik sinonimlashni ham avtomatik amalga oshirish uchun samarali yechimlarni izlash tadqiqot doirasida chuqurroq va uzoq izlanish lozimligini ko‘rsatadi.

Foydalanilgan adabiyotlar:

1. https://anexp.ru/articles/chto-takoe-sinonimaizer-texta-onlina-i-bez-poteri-smysla
2. https://www.similarweb.com/ru/website/synonymizer.ru/competitors
3. Uzoqova M. O‘zbek tili sinonimayzeri xususida about the synonymizer of the uzbek language// “O‘zbek tilining milliy korpusi: muammolar va vazifalar” xalqaro ilmiy-amaliy anjumani. Samarqand, 2023. – B. 288-292.
4. Wangchunshu Z. BERT-based Lexical Substitution. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. July, 2019. https://aclanthology.org/P19-1328.pdf
5. https://radimrehurek.com/gensim/auto_examples/tutorials/run_word2vec.html
6. https://en.m.wikipedia.org/wiki/Word2vec
7. https://www.kaggle.com/datasets/mohiyaxonuzokova/bigger-dataset-from-kunuz-and-daryouz

8. Jacob D., Ming-Wei Ch., Kenton L., and Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics. **DOI:** 10.18653/v1/N19-1423

9. Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692

10. https://huggingface.co/roberta-base

11. Baevski A., Edunov S., Liu Y., Zettlemoyer Y., Auli M. Cloze-driven Pretraining of Self-attention Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5360–5369, Hong Kong, China. Association for Computational Linguistics. **DOI:** 10.18653/v1/D19-1539

12. Elov B., Xusainova Z., Xudayberganov N. Tabiiy tilni qayta ishlashda Bag of Words algoritmidan foydalanish. O‘zbekiston: til va madaniyat (Amaliy filologiya), 2022, 5(4). 31-45.

13. https://huggingface.co/Mokhiya/syn-roberta