Poetry Generation Models

Eleen Kmail

NLP Course

Ms Yamama Shakaa

Introduction

Poetry generation is a fascinating field within Natural Language Processing (NLP) that aims to generate artistic and expressive poems using computational models. In this report, we will discuss the process and techniques used to train and evaluate two poetry generation models, one for Arabic poetry and the other for English poetry.

Data Collection

All the Poetries are collected from pre-collected datasets already published on the internet so in this project I did not collect them by my self.

For the Arabic poetry data set i utilized the "Arabic Poetry Dataset (6th - 21st century)" available on Kaggle. This dataset contains a wide range of Arabic poems from different time periods, the data set contains over 58K of poems , providing a rich source for training our model.
https://www.kaggle.com/datasets/fahd09/arabic-poetry-dataset-478-2017

For English poetry, we used the "Poem Dataset" provided by HCI Lab, which offers a diverse collection of English poems.
It consists of 199,002 verses, each of them is labeled with one of these four meters: Iambic, Trochee, Dactyl and Anapaestic.
https://hci-lab.github.io/LearningMetersPoems/

Models

Both Arabic and English models are based on GPT2 model , for the English model I fine-tuned the GPT2-base model as it is without beeing trained on poems before, for the arabic model I fine-tuned the arabic version of GPT2 which called aragpt2-base which is made specifically for arabic text .

Arabic model repo on Hugging Face: [Arabic Text Generator (aragpt2-base)]

English model repo on Hugging Face: [English Text Generator based on GPT2]

Embedding

In the process of generating poetry in both Arabic and English, I chose not to rely on transformer-based embeddings. Instead, I utilized sequence representation techniques for encoding the Arabic and English texts wich convert the text into vector of integer indexes assigned by a GPT2 tokenizer.

Then I prepare data as input and label data to fine-tune the models. The input is the whole sequence excluding the final word, and label is the whole sequence excluding the first word.

Libraries Used

During the poetry generation process, several libraries were utilized to enhance the functionality and efficiency of the models. The "transformer" library played a pivotal role in importing models from Hugging Face, allowing us to leverage the power of pretrained models specifically designed for NLP tasks. For sentence embedding and evaluation purposes, we employed the "SentenceTransformer" library, which facilitated the importation of models capable of encoding sentences into vector representations. The "cos_sim" function, another crucial component, enabled us to measure the similarity between two sentence embeddings, thereby evaluating the quality and coherence of the generated poems. Additionally, the "pyarabic.araby" library was utilized to remove diacritics from Arabic text, enhancing the models' ability to process and understand the linguistic nuances. Finally, the "tensorflow.keras.preprocessing.sequence" library played a vital role in preprocessing sequences, ensuring they were in the appropriate format for model training and generation. Collectively, these libraries significantly contributed to the successful implementation and evaluation of the poetry generation models.

Model Evaluation

For the evaluation of our poetry generation models, we employed Semantic Textual Similarity (STS) with Sentence-BERT pre-trained model. This approach involved embedding both the ground truth sentences (actual poems) and the predicted sentences generated by our models using the Sentence-BERT model. By encoding the sentences into vector representations, we were able to capture their semantic meaning and essence. Subsequently, we computed the cosine similarity between the embeddings of the ground truth and predicted sentences. This cosine similarity measure provided a quantitative evaluation of the semantic similarity between the generated poems and the desired poems, allowing us to assess the models' performance in generating poems that closely align with the given prompts. This evaluation method offered valuable insights into the quality and coherence of the generated poetic output.

Comparative Between Arabic and English Models

```
pd.set_option('display.max_colwidth', None)
output_df
```

[38]

	truth	predicted	score
11	وألثم ثغرها حكي اللآلي	وألثم ثغرها لحجرها من العين ، فجعلها	0.940099
7	كغصن البان في كُثب الرمالِ	كغصن البان بها	0.919654
6	وخصر يستبيك إذا تولت	.وخصر يستبيك لاكنه	0.877177
9	وجلّت كالمنيرة في الليالي	وجلّت كالمنيرة.أما أن تحصر معالجة من	0.870849
8	تبدّت كالقضيب على كثيب	تبدّت كالقضيبد من شدة العطش, نهضت من	0.856641
5	وفي أعماقها نبع الزلالِ	وفي أعماقها أعباها, هو مازال حيا بدمعته مشتاقا	0.853772
2	تميس فلا يعادلها قضيب	.(تميس فلا تضره	0.848158
10	فقمت أداعب الوجنات منها	فقمت أداعب ساقي بأذني حتى كاد أن ينتصب وجهي	0.839456
4	بمبسمها لَعَمرُ أبيك درٌّ	.بمبسمها لَعَمرُ	0.824243
12	وأهصرُ غصنها ضماً ولمّاً	وأهصرُ غصنها ولاقالهوآ عقى	0.792081
0	بدت تختال في حُلل الجمالِ	، بدت تختال خلف بعض الحيوانات التي لا ينتبه لها احد في الحديقة	0.784938
3	وإن ترنو تداعب بالنصال	وإن ترنو بعين الذاكرة إلى ما فيه خير الدنيا والآخرة ، فهيهات	0.776742
1	وجادت بالزيارة والوصال	وجادت بالزيارة الخاطفة التي قام بها وزير الخارجية الايراني علي اكبر صالحي	0.736243
14	ومن خلق العفاف لنا رقيب	....ومن خلق الله ، هذا الضوء الذي لا ينضب	0.679623
15	بطهر الحب في حسن الخلال	......بطهر الحب المصبوغ بفخر وفخر	0.657739
13	وألهو باليمين وبالشمال	وألهو باليمين ، ثم قلنا : " أما هذه ، فإنها لا	0.585546

```
pd.set_option('display.max_colwidth', None)
output_df
```

[18]

	truth	predicted	score
9	glares thro' the gloom and pours upon my breast	glares thro' all the joys of lifeand now weep'd the	0.600873
5	these sighs to murmur and these tears to flow	these sighs and pangs of grief and joyfellow of the common	0.575053
15	flame thro' my blood and steal me from my urn	flame thro' the grove of their headand charm'd every thought to	0.574694
11	and drags me back to misery and love	and drags me on the dusty road'ring mountain paths and amid the	0.540272
3	these earthborn visions saddening o'er my cell	these earthborn god are'dwith thy glowing and lightning hand they shall	0.491722
1	that wings my pulse and shoots from vein to vein	that wings and shall shew brighti on me with such a lustre	0.484887
10	bids heav'n's bright guard from paraclete remove	bids heav'n's softest motionhis roaring oars may calm	0.470400
6	'tis she 'tis eloisa's form restor'd	'tis she the true queen of truth and a poetess with a thousand	0.423425
12	enjoy thy triumphs dear illusion see	enjoy thy fruits the bestow'dof the birds and the trees for	0.328502
13	this sad apostate from his god to thee	this sad fate thro' my heart to that just landwhere the pain which	0.297925
14	see at thy call my guilty warmths return	see at last the caprice of dutyfor the sole cause of love and	0.286167
8	she comes in all her killing charms confest	she comes from abroad in camels or goats to teach thee the ways of	0.276593
0	ah why this boding start this sudden pain	ah why to the distant crywhere as you are no more ye shall be	0.273479
7	once a pure saint and more than saints ador'd	once a train in time to riseand safe for ever once will be the	0.211310
4	what strange disorder prompts these thoughts to glow	what strange yet still the savage's madours have ye not yet wilt	0.203696
2	what mean regardless of yon midnight bell	what mean to you to the brute and not to the brutelyfor the	0.128826

This score in the photos is the cosine similarity between the input and the predicted text after embedding them using S-BERT model, we can see that there is a high scores wich reflect that the model generate words that are close in the meaning to the ground truth or the topic of (seed test). from it we can see that the Arabic model has scores better than the English model for the majority of the samples so from this point we can say that the Arabic model perform good in the field of generating poems.

Human Interpretability Evaluation

Lets see some output of each model to talk about the poems generated.

Arabic model:

```python
seed_test= "الحزن"
input_ids = tokenizer.encode(seed_test, return_tensors='tf')
sample_outputs = model.generate(
input_ids,
do_sample=True,
max_length=15,
top_k=0,
top_p=0.9,
temperature=0.8,
num_return_sequences=3
)


# Print generated output
for i, output in enumerate(sample_outputs):
    print(f"Output {i + 1}: {tokenizer.decode(output, skip_special_tokens=True)}")
#  "chill'd the fair dawning of the yearin honour of th' auspicious morn",
```

```
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your input's `attention_m
Setting `pad_token_id` to `eos_token_id`:0 for open-end generation.
Output 1: الحزن وما للحزن عند الحزن فغنتها السنون تغنيها بعظ
Output 2: الحزن في القلب : لا شيء يزاده ويذويه في المحبوب كأنما
Output 3: الحزن لا يهن حتى ولو طال العمر طال الوصل أو طال الوصال أو
```

The seed text is "الحزن " and the model generate some lines related to the topic , we can see that the model connect the sadness with the heart in output 2 and output 3 is readably and has meaning , but the problem is theres no rhyme in generating the poems so the lines are not perfect.

English model:

```python
# Generate text
seed_test = "happy"
input_ids = tokenizer.encode(seed_test, return_tensors='tf')

sample_outputs = model.generate(
    input_ids,
    do_sample=True,
    max_length=100,
    top_k=0,
    top_p=0.9,
    temperature=1,
    num_return_sequences=5,
    pad_token_id=tokenizer.eos_token_id
)

# Print generated output
for i, output in enumerate(sample_outputs):
    print(f"Output {i + 1}: {tokenizer.decode(output, skip_special_tokens=True)}")
#  "chill'd the fair dawning of the yearin honour of th' auspicious morn",

```

[26] Python

```
Output 1: happy the birth of my redest lovewhat a matchless wind at their distant war
Output 2: happy every godly precept all the same lyreto whom all gladsome pleasure and paintoward that solemn solemn day
Output 3: happy britannia slain men his wonted herdson good the wise should say
Output 4: happy envy rewards on who oh no lie his yoke giveshe meets what haste and madness slumber
Output 5: happy a soul with no means for readingo and behold a shining light shine
```

Also in this model we can see in output 5 that he connect the happy with the soul word which they come related to each other in some contexts, also the model produce words like shine, love, good and pleasure which all related to the first seed text wich is happy.

Suggestions for performance improvements:

We searched for how we can make models to learn from data to get high score without overfitting happened, and we have reached to use a new loss function while training the model which uses similarities between the embeddings for the ground truth (actual) and predicted labels rather than the use the loss function which compare between the actual and predicted itself.

Models Deployment on Hugging Face Platform:

Link to Arabic model:
https://huggingface.co/EleenKmail/ArabicModelGenerator

Link to English model:
https://huggingface.co/EleenKmail/EnglishModelGenerator