



Аннотация

Для создания персонализированного ответа пользователю необходимо учитывать личную информацию о нем, заданный вопрос и доменные знания. Поэтому необходимо научиться извлекать релевантную информацию, которая поможет генеративной модели составить ответ пользователю. Для этого предлагается разбить данный процесс на три этапа: подбор релевантных предложений из базы знаний, подбор наиболее подходящих предложений персоны с учетом извлеченных знаний, генерация ответа на основе полученных данных. Для извлечения подходящих предложений из базы знаний и релевантных предложений персоны была использована модель на базе sentence-transformer all-mpnet-base-v2. Для генерации ответа была использована модель BART-base. Данная система достигла state-of-the-art результата на воркшопе The 1st Workshop on Customized Chat Grounding Persona.

Датасет

В ходе данных экспериментов использовался датасет FoCUS[1]. Данный датасет состоит из 14,452 диалогов. Структура диалога и пример диалога представлены ниже.

```
{
# список фактов об объекте
"knowledge": List[str],
# 5 предложений персоны
"persona": List[str],
# диалог, 5-7 итераций
"utterance": [
{
# вопрос и ответ
"dialogue1": List[str],
# метки используемой персоны
"persona_grounding": List[bool],
# копия предложений из персоны 5шт
"persona_candidate": List[str],
# кандидаты на знание, 1 правильный, 9 неправильных
"knowledge_candidates": List[str],
# индекс правильного знания
"knowledge_answer_index": int,
},
],
}
```

Human
Wow, this is amazing! What is this?

Machine(personalized answer)
This is the Great Pyramid of Giza. As you would like to visit all of the Seven Wonders of the Ancient World, you will be excited to hear that it is oldest, and most intact, of these wonders.

Knowledge: The Great Pyramid of Giza is the oldest and largest... of Seven Wonders of the Ancient World, ... and most intact

Persona: I would like to visit all of Seven Wonders of the Ancient World.

Knowledge Extraction

На каждой итерации диалога необходимо подавать в модель релевантную информацию, на основе которой будет выполнена генерация ответа. Данную задачу решает ранжирующая модель sentence-transformers/all-mpnet-base-v2[2], обученная под данный датасет. Во время обучения использовался Symmetric Cross Entropy Loss, аналогичный тому, что был представлен в работе CLIP[3]. В качестве метрики близости предложений использовалось косинусное расстояние между векторными представлениями предложений. Псевдокод вычисления приведен ниже:

```
source_sentences = lm_model(context)
knowledge_sentences = lm_model(knowledge)
labels = range(knowledge.length)
scale_factor = 20
scores = source_sentences * knowledge_sentences *
scale_factor
loss_f = CrossEntropyLoss()
loss = (loss_f(scores, labels) + loss_f(scores.T, labels))/2
```

context - сконкатенированная целиком персона и последний вопрос пользователя. **knowledge** - знание которое использованное для ответа в датасете. **scale_factor** - гиперпараметр существенно влияющий на сходимость.

Persona Extraction

Для генерации ответа необходимо определить какие личные данные о пользователе уместно использовать в данном контексте. В среднем при генерации ответа персону используют только в 13% случаях. Поэтому мы имеем дело с имбалансным датасетом. Задача была сформулирована как бинарная классификация. Обучение происходило при помощи модели sentence-transformers/all-mpnet-base-v2 и функции потерь FocalLoss[4] с гиперпараметрами gamma=2.0, alpha=0.5. Элементы для классификации формировались при помощи конкатенации предложения из персоны, предложения с доменными знаниями и последнего вопроса пользователя.

Response Generation

Чтобы ответить на вопрос пользователя данная система сначала извлекает наиболее релевантные доменные знания. Затем, на основе извлеченных знаний, система определяет какие предложения из персоны можно использовать для генерации ответа. Наконец, полученная информация вместе с последним вопросом от пользователя подается в BART-base[5]. Общая схема данной системы приведена ниже:



Результаты

Результаты экспериментов из секций Knowledge и Persona Extraction:

Knowledge Extraction Experiments	Knowledge Accuracy	Dataset	Persona Extraction Experiments	Persona Accuracy	Dataset
debertav3-base+Binary Cross-Entropy	48.15	valid	predict only zeros(never use persona)	86.69	valid
TF-IDF	61.23	valid	debertav3-small+Cross-Entropy Loss	92.12	valid
sentence-transformers/all-mpnet-base-v2	93.43	valid	sentence-transformers/all-mpnet-base-v2+Cross-Entropy Loss	92.44	valid
sentence-transformers/all-mpnet-base-v2	93.38	test	debertav3-small+Weighted Cross-Entropy Loss	91.74	valid
finetuned sentence-transformers/all-mpnet-base-v2	98.9	valid	debertav3-small+Weighted Cross-Entropy Loss	91.4975	test
finetuned sentence-transformers/all-mpnet-base-v2	98.8467	test	sentence-transformers/all-mpnet-base-v2+focal loss	92.45	valid
finetuned debertav3-small	17.68	valid	sentence-transformers/all-mpnet-base-v2+focal loss	92.2711	test
debertav3-small	7.98	valid	debertav3-small+focal loss	92.12	valid

В обеих задачах лучше всего показала себя модель sentence-transformers/all-mpnet-base-v2. В результате данная система превзошла изначальное решение авторов и взяла первое место на воркшопе The 1st Workshop on Customized Chat Grounding Persona. Предложенный метод обозначен в таблице как deepavlov, focus-nc решение предложенное авторами, anonymous - топ-2 решение другой команды.

User	Persona Extraction	Persona Accuracy	Knowledge Extraction	Knowledge Accuracy	Generative Model	SacreBLEU	CharF++	ROUGE-L
focus-nc	BART-classifier	86.8595	BART-classifier	65.0639	BART-base	10.8747	27.9026	30.9892
Anonymous	-	87.1434	-	90.5252	-	25.085	46.6757	47.0984
deepavlov	sentence-transformers/all-mpnet-base-v2+focal loss	92.2711	sentence-transformers/all-mpnet-base-v2	98.8467	BART-base	32.4795	53.9012	54.3133

References: [1] arXiv:2112.08619, [2] arXiv:1908.10084, [3] arXiv:2103.00020, [4] arXiv:1910.02190v2, [5] arXiv:1910.13461

