# Milestone 2 Report

## Section 4: LLaMA Inference

### 4.3 Model Architecture

The KV cache structure is used to accelerate inference. It is removed according to the requirement.

```
# self.cache_k = self.cache_k.to(xq)
# self.cache_v = self.cache_v.to(xq)

# self.cache_k[:bsz, start_pos : start_pos + seqlen] = xk
# self.cache_v[:bsz, start_pos : start_pos + seqlen] = xv

# keys = self.cache_k[:bsz, : start_pos + seqlen]
# values = self.cache_v[:bsz, : start_pos + seqlen]
keys = xk
values = xv
```

Also modified the code for mask to adapt the change

```
# mask = torch.triu(mask, diagonal=start_pos + 1).type_as(h)
mask = torch.triu(mask, diagonal=1).type_as(h)
```

Apart from deleting code related to this part in `model.py` , we also modify part of the code in `generation.py` as below:

```
for cur_pos in range(min_prompt_len, total_len):

    logits = self.model.forward(tokens[:, 0:cur_pos], prev_pos)
```

in order to fetch all token values at one time.

For `fairscale.nn.model parallel.layers` , we replace them with corresponding `nn.Linear` layers.

```
# Attention
self.wq = nn.Linear(args.dim, args.n_heads * self.head_dim, bias=False)
self.wk = nn.Linear(args.dim, args.n_heads * self.head_dim, bias=False)
self.wv = nn.Linear(args.dim, args.n_heads * self.head_dim, bias=False)
self.wo = nn.Linear(args.n_heads * self.head_dim, args.dim, bias=False)
# FeedForward
self.w1 = nn.Linear(dim, hidden_dim)
self.w2 = nn.Linear(hidden_dim, dim)
self.w3 = nn.Linear(dim, hidden_dim)
# Transformer
self.tok_embeddings = nn.Embedding(params.vocab_size, params.dim)

self.output = nn.Linear(params.dim, params.vocab_size, bias=False)
```

## 4.5 Run Inference

To generate text by directly running `python -m ...` , we also disable distributed features. Specifically, we hardcode the environment variables in `generation.py` :

```
# Set environment variables for distributed training
os.environ['RANK'] = '0'
os.environ['WORLD_SIZE'] = '1'
os.environ['MASTER_ADDR'] = 'localhost'
os.environ['MASTER_PORT'] = '12345'
```

This is the command we use to run inference:

```
python example_text_completion.py --ckpt_dir /project/saifhash_1190/llama2-7b
        --tokenizer_path /project/saifhash_1190/llama2-7b/tokenizer.model
```

We test with some new texts.

### Zero shot prompts

```
What is machine learning?
> How does it work?
A lot of people have heard about machine learning, but few know what it actually is.
Machine learning is a type of artificial intelligence (AI) that allows computers to
learn from data without being explicitly programmed.
Machine learning algorithms are used to make predictions based on patterns in data. For
```

========================================

```
To maintain a healthy work-life balance, it's essential to
> have a healthy lifestyle. It is not always easy to maintain a healthy lifestyle,
especially when you have a busy schedule. However, it is important to make time
for yourself and your health. This article will discuss how to maintain a healthy
work-life balance with a healthy lifest
```

========================================

## Few shot prompts

```
Find synonyms for the following words:
        quick => fast
        happy => joyful
        difficult => challenging
        intelligent =>
> smart
        silly => dumb
        simple => easy
        fast => quick
        loud => noisy
        dumb => silly
        scary => spooky
        noisy => loud
        quiet => silent
        smart => intelligent
        challenging => difficult
        easy =>
```

========================================

```
 Match the event to the year:

         Moon Landing => 1969
         Fall of the Berlin Wall => 1989
         Invention of the Internet => 1983
         Start of World War I =>
 > 1914
         Start of World War II => 1939
         Start of the Vietnam War => 1964
         Start of the Korean War => 1950
         Start of the Cold War => 1947
         Start of the Cuban


========================================
```