Research on DocFormer

1. Introduction

DocFormer is a Transformer-based document understanding model designed to process textual, layout, and visual information in a unified end-to-end architecture. It was introduced in the paper titled "DocFormer: Endto-End Transformer for Document Understanding".

This research explores the process of fine-tuning DocFormer for extracting structured information from financial documents such as invoices, receipts, and bank statements.

2. Objective

The goal is to fine-tune the DocFormer model on datasets like FUNSD or annotated financial documents to extract entities like date, total amount, and vendor name using token classification.

3. Resources Used

- GitHub Repository: https://github.com/shabie/docformer (Unofficial mirror)
- FUNSD Dataset for Named Entity Recognition and document layout analysis
- Transformers (HuggingFace) and PyTorch for model training
- Label Studio for annotation and visualization

Research on DocFormer

4. Step-by-Step Attempt to Set Up DocFormer

1. **Cloned the GitHub repo shabie/docformer**
    - Repo cloned into the inner shabie-docformer/ directory inside the main working folder.
    - Data folder contain the Funsd Dataset.
2. **Converted FUNSD dataset into train.txt**
    - Tokens, bounding box coordinates, and BIO labels extracted and written line-by-line.
    - Format: word<TAB>xmin ymin xmax ymax<TAB>label, separated by double newline per sample.
3. **Created funsd_dataset.py**
    - Custom FunsdDataset class implemented using torch.utils.data.Dataset.
    - Purpose: Read train.txt, apply LayoutLM/DocFormer tokenizer, align tokens with bounding boxes and labels.
    - Output dictionary includes: input_ids, attention_mask, bbox, token_type_ids, labels.
    - New docformer_outputs folder created when run funsd_dataset.py
4. **Created train_docformer.py**
    - Initial experiments were with microsoft/layoutlm-base-uncased.
    - Later modified to attempt fine-tuning DocFormerForTokenClassification from the shabie/docformer repo.
    - Used HuggingFace Trainer with arguments like batch size, logging steps, output directory, etc.
5. **Set up model initialization attempts:**
    - Imported model from docformer.modeling (custom implementation).
    - from_pretrained() call used config.json and attempted to load weights.
    - Manually added missing config fields (intermediate_ff_size_factor, etc.) to prevent KeyErrors.
6. **Challenges with docformer_outputs:**
    - This directory was intended to store checkpoints.
    - However, only training_args.bin was saved because training didn't complete successfully.
    - Crucial file like pytorch_model.bin was missing, resulting in multiple runtime failures during model loading.
7. **Observed errors and debugging:**
    - Tracked issue to the fact that shabie/docformer weights are **not publicly available** on HuggingFace.
    - The repo lacks pre-trained .bin files and proper hosting of tokenizer/model configs.

8. **Verified that shabie/docformer HuggingFace links are broken:**
    - URL attempts like: https://huggingface.co/shabie/docformer/resolve/main/config.json return 404 or 401.
    - Model is not visible via public model search on HuggingFace.

5. Challenges Faced

- Import Errors due to incorrect module structure: Resolved using path adjustments.

- Missing keys in config.json: Added `intermediate_ff_size_factor`, `max_relative_positions`, etc.

- No official weights available: The repository `shabie/docformer` does not provide any pretrained weights.

- All links in the repo (e.g., Hugging Face links) are broken or private and cannot be accessed publicly.

- PyTorch Unpickling errors: Due to `training_args.bin` being incorrectly used as weight files.

- Final error: Absence of `pytorch_model.bin` makes training from checkpoint impossible.

6. Current Status

Due to the absence of pretrained weights and the lack of functional links in the `shabie/docformer` repository, we have halted active setup and training efforts.

The model architecture and dataset parsing logic are working, but fine-tuning cannot proceed without pretrained weights. We are awaiting public release or access to pretrained `pytorch_model.bin` to resume.

7. References

- shabie/docformer GitHub: https://github.com/shabie/docformer

- FUNSD Dataset: https://guillaumejaume.github.io/FUNSD/

- Paper: "DocFormer: End-to-End Transformer for Document Understanding" (Shalini Ghosh et al.)