--- Interactive FAISS + DeepSeek Chat Query ---

Type 'exit' to quit.

Enter your query: 发票中的内容是什么

--- Response ---

根据提供的文档片段，无法直接提取出完整的发票内容。这些片段主要包含的是PDF文件中的字体定义、编码信息、对象引用和资源描述等元数据，例如：

1. 字体描述（如Helvetica、Times New Roman的字符编码、宽度、边界框等）

2. PDF内部对象的结构（如`/Type /Font`、`/Encoding`、`/CharProcs`等）

3. 部分乱码或编码的流数据（如`x = 0C @ ) |O j #K...`等不可读内容）

4. 图形文件引用（如`./figs/consistency\_power.pdf`）

\*\*可能的相关信息\*\*：

- 文档中出现的字符集包含常见符号（如`space`、`parenleft`、`hyphen`）和字母（如`a`-`z`、`A`-`Z`）以及数字（`zero`-`nine`）。

- 部分片段提到`Matplotlib`生成的PDF，可能包含图表或图形。

\*\*结论\*\*：

这些片段是PDF文件的内部技术数据，未显示具体的发票文本内容（如收款方、金额、日期等）。如果需要解析发票内容，建议：

1. 直接查看PDF的可见页面文本层。

2. 使用PDF文本提取工具（如`pdftotext`）。

3. 检查文档中是否有`/Contents`或文本流对象（未在片段中提供）。

Enter your query: Empirical Study of Zero-Shot NER with ChatGPT论文的github仓库链接是什么

--- Response ---

根据提供的文档片段，没有直接提到《Empirical Study of Zero-Shot NER with ChatGPT》这篇论文的GitHub仓库链接。文档中虽然包含了一些与论文相关的元数据和引用信息（如`cite.RepoCoder`、`cite.CodeAgent`等 ），但并未明确关联到该特定论文的代码仓库。

建议您通过以下方式查找：

1. 直接搜索论文标题 + "GitHub"。

2. 检查论文的官方发布渠道（如ACL Anthology、arXiv等）的"Resources"部分。

3. 联系论文作者获取信息。

如果需要进一步帮助定位，可以提供更完整的论文信息或相关上下文。

Enter your query: Empirical Study of Zero-Shot NER with ChatGPT + "GitHub"

--- Response ---

The provided document fragments appear to be technical PDF metadata or font descriptors, which do not contain any substantive information about an empirical study on \*\*Zero-Shot Named Entity Recognition (NER) with ChatGPT\*\* or its application to \*\*GitHub-related data\*\*.

To answer your question properly, here’s a general overview based on existing research trends (since the fragments don’t contain relevant content):

### \*\*Zero-Shot NER with ChatGPT\*\*

1. \*\*Definition\*\*: Zero-shot NER involves identifying entities (e.g., person, organization) in text without task-specific training data, relying on a model's pre-existing knowledge.

2. \*\*ChatGPT’s Performance\*\*:

- Studies (e.g., [Li et al., 2023](https://arxiv.org/abs/2305.15254)) show that ChatGPT can perform reasonably in zero-shot NER but struggles with fine-grained entity types or domain-specific terms (e.g., software repositories).

- Challenges include \*\*inconsistent entity boundaries\*\* and \*\*hallucinations\*\* (fabricated entities).

3. \*\*GitHub-Specific Applications\*\*:

- For GitHub (e.g., issue reports, commit messages), zero-shot NER might identify:

- \*\*Code-related entities\*\* (e.g., `Library/Framework`, `API`, `ErrorType`).

- \*\*Developer discussions\*\* (e.g., `FeatureRequest`, `BugReport`).

- However, performance drops without fine-tuning due to jargon (e.g., `PyTorch`, `CUDA`).

### \*\*Key Findings from Empirical Studies\*\*

- \*\*Strengths\*\*:

- Leverages ChatGPT’s broad pretraining to generalize across domains.

- Useful for prototyping or low-resource scenarios.

- \*\*Limitations\*\*:

- \*\*Precision/Recall Trade-offs\*\*: High false positives in technical contexts.

- \*\*Prompt Sensitivity\*\*: Performance varies heavily with prompt engineering (e.g., "List all software libraries in this text" vs. "Extract entities").

### \*\*Recommendations\*\*

- \*\*Hybrid Approaches\*\*: Combine ChatGPT with rule-based post-processing (e.g., regex for GitHub-specific patterns).

- \*\*Few-Shot Learning\*\*: Provide 2–3 examples in prompts to improve accuracy.

If you’re looking for a specific study, I recommend searching arXiv or ACL Anthology for papers like \*"Evaluating ChatGPT on Zero-Shot Information Extraction"\* or \*"NER for Software Documentation"\*. Let me know if you’d like help refining the query!

\*(Note: The provided document fragments did not contain usable information—this answer synthesizes general knowledge.)\*

(RAG\_DEMO) E:\RAG>python data\_ocr.py

Using CPU. Note: This module is much faster with a GPU.

Downloading detection model, please wait. This may take several minutes depending upon your network connection.

Progress: |██████████████████████████████████████████████████| 100.0% CompleteDownloading recognition model, please wait. This may take several minutes depending upon your network connection.

Progress: |██████████████████████████████████████████████████| 100.0% CompleteProcessing knowledge\29041d3a-92f1-41de-9158-f710cb274013.jpg ...

Processing knowledge\发票.jpeg ...

Processing knowledge\LLM\_vote.pdf ...

Token indices sequence length is longer than the specified maximum sequence length for this model (319072 > 512). Running this sequence through the model will result in indexing errors

Processing knowledge\sample.pdf ...

Processing knowledge\NeurIPS-2022-training-language-models-to-follow-instructions-with-human-feedback-Paper-Conference.pdf ...

Built index with 3256 chunks.

Index saved to knowledge\_ocr\_chunks.index

Metadata saved to knowledge\_ocr\_chunks\_meta.pkl

(RAG\_DEMO) E:\RAG>python query\_ocr.py

--- Interactive FAISS + DeepSeek Chat Query ---

Type 'exit' to quit.

Enter your query: 西华大学在哪里开了一张增值税发票？

--- Response ---

根据提供的文档片段，可以提取出以下关于西华大学增值税发票的信息：

1. \*\*发票抬头\*\*：显示为"西华犬^"（疑似OCR识别错误，应为"西华大学"）

2. \*\*纳税人识别号\*\*：12510000450717578

3. \*\*开票内容\*\*：

- 项目经济性分析与评价（金额35.50元）

- 贷款项目可行性分析（金额44.30元）

- 合计价税金额：79.8元

4. \*\*销售方信息\*\*：

- 名称：当当网信息技术（四川）有限公司

- 纳税人识别号：91511422462375298

- 地址：四川省成都市武侯区（片段中地址部分文字模糊）

\*\*结论\*\*：

该增值税发票是由\*\*当当网信息技术（四川）有限公司\*\*向\*\*西华大学\*\*开具的，发票内容涉及图书或咨询服务类项目。由于文档片段存在文字识别错误和格式混乱，完整地址等细节需进一步核实原始票据。建议通过西华大学财务部门或国家税务总局发票查验平台（需输入发票代码和号码）查询完整信息。

（注：片段中部分字段如地址、开户行等信息因格式错乱无法完整提取）