



## (File)

- 文件 : 包含 数据 的 集合 。
- 文件 :
  - 文件 名 (文件 名 ) 包含 文件 名 。
  - 文件 名 , 文件 名 , 文件 名 包含 文件 名 。
  - 文件 名 包含 文件 名 。



## (Target)

- 目标 : 包含 数据 的 集合 。
- 目标 :
  - 目标 名 包含 目标 名 。
  - 目标 名 包含 目标 名 / 目标 名 。
  - 目标 (Source) 包含 目标 名 , 目标 名 。



## (Source)

- 源 : 包含 数据 的 集合 。
- 源 :
  - 源 名 包含 源 名 。
  - 源 名 包含 源 名 。
  - 源 名 包含 源 名 。
  - 源 名 包含 源 名 。



## (Dataset)

- 数据集 : 包含 数据 的 集合 。
- 数据集 :
  - 数据集 (Source Dataset): 包含 数据 的 集合 。

- Source, "Source" .
- Target Dataset: .
- Target, "Target" .

## Token

- Token: .
- Token types:
  - Word token: .
  - Subword token (Byte Pair Encoding):
    - Word token: .
    - Subword token: ( ) .
    - Subword token: , .
    - Subword token: , .
    - Subword token: .

## Sequence

## (Subsequence)

- Sequence:
  - Sequence: .
  - Subsequence: .

## Minimum Token Sequence, MTS

## (Minimum Token Sequence Unit, MTSU)

- MTS:
  - MTS: .
  - MTSU: , , .
- MTSU:
  - MTSU MTS .
  - MTSU MTS .

# [[ ]] (Tokenized File)

- [[ ]]: [[ ]] [[ ]] [[ ]]
- [[ ]]:
  - [[ ]] [[ ]]
  - [[ ]] [[ ]] [[ ]] [[ ]]:
    - [[ ]]
    - [[ ] [[ ] ( [[ ] ) ]]

# [[ ] (Query), [[ ] (Query Request), [[ ] (Query Result)

!!TODO: [[ ] (Candidate Query) [[ ] (Query) [[ ] [[ ]

- [[ ]:
  - [[ ] **(Query)**: [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] **(Query Request)**: [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] **(Query Result)**: [[ ] [[ ] , [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
- [[ ] [[ ]:
  - [[ ] [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
- [[ ]:
  - [[ ] [[ ] **(Exact Query)**: [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] **(Similar Query)**: [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]
- [[ ]:
  - [[ ] [[ ] [[ ] [[ ] : [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] [[ ] : [[ ] [[ ] [[ ] [[ ] [[ ] [[ ]

# [[ ] (Candidate Query)

!!TODO: [[ ] (Candidate Query) [[ ] (Query) [[ ] [[ ]

- [[ ]: [[ ] [[ ] [[ ] [[ ] [[ ] , [[ ] [[ ] [[ ]
- [[ ]:
  - [[ ] [[ ] [[ ] [[ ]
  - [[ ] [[ ] [[ ] [[ ] [[ ]



- ID를 포함한 모든 문자를 포함합니다.
- 모든 문자를 포함합니다.

## Indexer (Indexer)

- Indexer : 모든 파일을 인덱싱합니다.
- Indexer :
  - \*\*Source Files\*\* :
    - \*\*Source Files\*\* \*\*Tokenized Files\*\* .
    - \*\*Tokenized Files\*\* \*\*MTSU\*\* 인덱싱합니다.
  - \*\*Index\*\* .

## Querier (Querier)

- Querier : \*\*Target Files\*\* 인덱싱합니다.
- Querier :
  - \*\*Target Files\*\* :
    - \*\*Target Files\*\* \*\*Tokenized Files\*\* .
    - \*\*Tokenized Files\*\* **MTSU** 인덱싱합니다.
    - **MTSU** 인덱싱합니다.
    - 인덱싱합니다 / **MTSU**.

## Finder (Finder)

- Finder : 모든 파일을 찾습니다.
- Finder :
  - **MTSU** 인덱싱합니다 (ID) \*\*Index\*\* .
  - 인덱싱합니다 .

## Merger (Merger)

- Merger : 모든 파일을 병합합니다.
- Merger :
  - 인덱싱합니다 .
  - 인덱싱합니다 .

## Hash Function (Hash Function)

- **MTS** : **MTSU** (Minimum Search Unit)의 크기를 지정하는 데 사용된다.
- **FarmHash** :
  - FarmHash는 MTS를 사용하여 유사도를 계산하는 데 사용된다.
  - FarmHash는 MTS의 크기를 지정하는 데 사용된다.

## 언어 (Language Family)

- **언어** : 언어의 유사도를 계산하는 데 사용된다.
- **언어** :
  - 언어는 유사도를 계산하는 데 사용된다.
  - 언어는 유사도를 계산하는 데 사용된다.
  - 언어는 유사도를 계산하는 데 사용된다.

## 구분 (Partitioning)

- **구분** : 유사도를 계산하는 데 사용된다.
- **구분** :
  - 구분은 유사도를 계산하는 데 사용된다.
  - 구분은 유사도를 계산하는 데 사용된다.
  - 구분은 유사도를 계산하는 데 사용된다.

## 매개변수 (Parameter)

- **매개변수** : 유사도를 계산하는 데 사용된다.
- **매개변수** :
  - **Similarity Rate**: 유사도를 계산하는 데 사용된다.
  - **MTSU Size** (Minimum Search Unit Size):
    - MTSU Size는 유사도를 계산하는 데 사용된다.
    - MTSU Size는 유사도를 계산하는 데 사용된다.

## 프로젝트 요구사항 (Project Requirements)

- **MTS** :
  - MTS는 3 이상이어야 한다.

- MTS 64KB 64KB 64KB .
- 内存池 :
  - 内存池 X KB X KB (内存池 内存池 ) .
- 内存池 内存池 :
  - 内存池 内存池 内存池 , 内存池 内存池 .
  - $2^{36}$  内存池 内存池 , 内存池 内存池 .
- 内存池 内存池 :
  - Roaring Bitmap 内存池 内存池 , 内存池 内存池 内存池 .
- 内存池 内存池 :
  - 内存池  $2^{16}$   $2^{24}$  MTS 内存池 内存池 .
  - 内存池 内存池 内存池 内存池 , 内存池 内存池 .
  - 内存池 10% 内存池 90% 内存池 内存池 .
- 内存池 内存池 :
  - 内存池 内存池 内存池 内存池 内存池 .
  - RocksDB/LevelDB MultiGet API 内存池 内存池 API 内存池 .
- 内存池 内存池 :
  - 内存池 内存池 (内存池 内存池 ) 内存池 8KB 内存池 (SaaS 内存池 ) 内存池 .
  - 内存池 内存池 内存池 内存池 内存池 内存池 .

## 内存池 (Hardware)

- 内存池 : 内存池 内存池 内存池 内存池 内存池 .
- 内存池 :
  - 内存池 内存池 : 内存池 内存池 内存池 内存池 内存池 .
  - 内存池 内存池 : SaaS 内存池 内存池 内存池 内存池 .
  - SSD 内存池 内存池 内存池 内存池 .
  - 内存池 内存池 内存池 内存池 内存池 内存池 .

## 内存池 内存池 (Hashing and Key Distribution)

- 内存池 : 内存池 内存池 内存池 内存池 内存池 .
- 内存池 :
  - 内存池 内存池 内存池 内存池 .
  - 内存池 内存池 内存池 内存池 内存池 .
  - 内存池 内存池 内存池 内存池 .

## MultiGet API

- 内存池 : RocksDB LevelDB 内存池 内存池 API .



○ SSD 性能 改善 方法 について 説明 します。

## SSD 最適化 (SSD Optimization)

- SSD : SSD (Solid State Drive) は 従来の HDD と 異なり 高速 かつ 低消費電力 で 動作 します。
- SSD の 最適化 方法 :
  - TRIM 機能 を 有効 に する ことで SSD の 性能 を 向上 させ ます。
  - SSD の 使用 容量 を 80% 未満 に 保つ こと が 最適 です。
  - SSD の 寿命 を 延ばす ため に 定期的 に 温度 管理 を 行い ます。

## ユニフォームキー分布 (Uniform Key Distribution)

- 問題 : データ が 特定の キー に 集中 すると SSD の 寿命 が 短縮 され ます。
- 解決 策 :
  - FarmHash など の ハッシュ 関数 を 利用 して キー を 分散 させ ます。
  - データ を 定期的に 再配置 して キー の 分布 を 均一 に 保つ ます。
  - キー の 重複 を 検出 し 除去 して 分布 を 改善 させ ます。

この 文書 は SSD の 性能 を 最大化 し 寿命 を 延ばす ための 最適化 方法 と ユニフォーム キー 分布 の 重要性 について 説明 します。 具体的な 設定 方法 と 注意点 を 詳細 に 記載 して います。

Revision #1

Created 2025-01-06 15:08:34 UTC by PPUZZL

Updated 2025-01-06 15:18:40 UTC by PPUZZL