Scraping Telegram

Francesco Pinsone

Indice

.

1 Introduzione

L'obiettivo è quello di estrarre dati sensibili su possibili minacce informatiche da Telegram. La suite di script implementati effettua un'attività di scraping su vari canali target di Telegram. Ciò viene realizzato tramite l'utilizzo della libreria `Telethon`, libreria ufficiale che consente l'accesso alle api di Telegram agli utenti registrati. Il software prende in input una lista di canali target, di cui l'utente deve necessariamente essere membro. In uscita si ottiene invece che i dati estratti vengono organizzati in oggetti JSON che vengono poi salvati su un apposito database MongoDB.

2 Scraping

Per fare scraping di contenuti da Telegram, è possibile utilizzare la libreria Python `Telethon`, che consente di interagire con le API di Telegram, accedere ai messaggi dei canali pubblici o ai canali in cui l'utente è registrato e filtrare le informazioni rilevanti. A differenza di tecniche di scraping basate su browser o su librerie come `Selenium`, `Telethon` permette un'interazione diretta con i server di Telegram tramite l'uso delle API ufficiali, offrendo così una modalità più stabile e conforme ai termini di utilizzo della piattaforma.

Attraverso le API di Telegram, un programma di scraping può accedere ai messaggi di specifici canali utilizzando le credenziali dell'utente o del bot, configurate tramite `api_id` e `api_hash`. Una volta connesso, lo script può accedere alla cronologia dei messaggi di un canale e filtrare i contenuti in base a parole chiave specificate, estraendo informazioni strutturate per analisi mirate. Durante questo processo, i messaggi vengono trattati per rimuovere campi non necessari o dati sensibili, e successivamente salvati in un database per garantire una consultazione e un'analisi future.

Questa tecnica di scraping risulta vantaggiosa in contesti di ricerca e monitoraggio, in quanto consente di accedere e analizzare contenuti pubblici o su invito, mantenendo alta la qualità e l'affidabilità dei dati estratti.
Per una descrizione più approfondita del codice sorgente si rimanda alla documentazione dettagliata.

3 MongoDB

MongoDB è un database NoSQL orientato ai documenti, progettato per gestire grandi quantità di dati non strutturati o semi-strutturati. A differenza dei database relazionali tradizionali (RDBMS), MongoDB non utilizza tabelle e schemi rigidi, ma archivia i dati in documenti JSON-like chiamati *BSON* (Binary JSON), che permettono una maggiore flessibilità nella gestione delle informazioni.
MongoDB è organizzato in:

- **Database**: è l'unità più grande che contiene una collezione di dati correlati.
- **Collezioni**: ogni database può contenere una o più collezioni, l'equivalente delle tabelle in un database relazionale. Tuttavia, una collezione non ha uno schema fisso.
- **Documenti**: all'interno di una collezione, i dati vengono memorizzati sotto forma di documenti BSON. Ogni documento è una struttura JSON-like che contiene campi chiave-valore. Questa flessibilità permette che i documenti all'interno della stessa collezione possano avere campi differenti.

MongoDB supporta le operazioni CRUD (Create, Read, Update, Delete) e permette la memorizzazione di dati complessi, come array o documenti nidificati. È particolarmente adatto per applicazioni che devono gestire dati eterogenei o in rapido cambiamento, come il web scraping o i big data.
Nel codice Python, MongoDB viene utilizzato tramite la libreria `pymongo`, che consente di interagire facilmente con il database.

4 Risultati

Una volta eseguita la ricerca, effettuato lo scraping e il parsing dei dati estratti questi ultimi vengono salvati su MongoDB. Prima di essere salvati, i risultati vengono organizzati, durante il processo di parsing, in oggetti JSON che poi potranno essere archiviati direttamente.
Il formato degli oggetti salvati è il seguente:

```
{
    "id": "int",
    "peer_id": "object",
    "date": "datetime",
    "mentioned": "boolean",
    "post": "boolean",
    "reply_to": "string",
    "media": "object",
    "entities": "array",
    "views": "int",
    "replies": "int",
    "edit_date": "datetime",
    "post_author": "string",
    "reactions": "int"
}
```

Esempio:


```
{
  "id": { "$numberInt": "2219" },
  "peer_id": {
    "_": "PeerChannel",
    "channel_id": { "$numberInt": "1210717532" }
  },
  "date": 2024-07-26T15:10:06.000+00:00
  "message": "In bocca al lupo agli atleti italiani alle Olimpiadi..",
  "mentioned": false,
  "post": true,
  "reply_to": null,
  "media": {
    "_": "MessageMediaPhoto",
    "spoiler": false,
    "photo": {
      "_": "Photo",
      "id": {
        "$numberLong": "5846199193306253179"
      },
```

```
"access_hash": {
  "$numberLong": "-7687017670132623686"
},
"file_reference": {
  "$binary": {
    "base64": "AkgqFVwAAAirZt7/yLCAY3jDeZOZjxRX5rIhOI8=",
    "subType": "00"
  }
},
"date": {
  "$date": {
    "$numberLong": "1722006606000"
  }
},
"sizes": [
  {
    "_": "PhotoStrippedSize",
    "type": "i",
    "bytes": {
      "$binary": {
        "base64": "ASgopqMsKkkjwQANvfk1GCQcjtT5pTMwO3GAB+lba3MFYYRj0/A1KRF5ZGOdoO739Kh
        "subType": "00"
      }
    }
  },
  {
    "_": "PhotoSize",
    "type": "m",
    "w": { "$numberInt": "320" },
    "h": { "$numberInt": "320" },
    "size": { "$numberInt": "25056" }
  },
  {
    "_": "PhotoSize",
    "type": "x",
    "w": { "$numberInt": "800" },
    "h": { "$numberInt": "800" },
    "size": { "$numberInt": "95530" }
  },
  {
    "_": "PhotoSizeProgressive",
    "type": "y",
    "w": { "$numberInt": "1080" },
    "h": { "$numberInt": "1080" },
    "sizes": [
      { "$numberInt": "9984" },
```

```
            { "$numberInt": "22848" },
            { "$numberInt": "35777" },
            { "$numberInt": "48302" },
            { "$numberInt": "92065" }
          ]
        }
      ],
      "dc_id": { "$numberInt": "4" },
      "has_stickers": false,
      "video_sizes": []
    },
    "ttl_seconds": null
  },
  "entities": [
    {
      "_": "MessageEntityHashtag",
      "offset": { "$numberInt": "284" },
      "length": { "$numberInt": "10" }
    }
  ],
  "views": 151893,
  "forwards": { "$numberInt": "38" },
  "replies": null,
  "edit_date": null,
  "post_author": null,
  "reactions": null
}
```


5 Repository Github

Di seguito il link alla Repository Github.