tinybird

Analyze API

The Analyze API allows you analyze a given NDJSON or Parquet file to generate a Tinybird Data Source schema.

POST /v0/analyze/?

The Analyze API takes a sample of a supported file (`csv`, `ndjson`, `parquet`) and guesses the file format, schema, columns, types, nullables and JSONPaths (in the case of NDJSON paths).

This is a helper endpoint to create Data Sources without having to write the schema manually.

Take into account Tinybird's guessing algorithm is not deterministic since it takes a random portion of the file passed to the endpoint, that means it can guess different types or nullables depending on the sample analyzed. We recommend to double check the schema guessed in case you have to make some manual adjustments.

ANALYZE A LOCAL FILE

```
curl \
-H "Authorization: Bearer <DATASOURCES:CREATE token>" \
-X POST "https://api.tinybird.co/v0/analyze" \
-F "file=@path_to_local_file"
```

ANALYZE A REMOTE FILE

```
curl \
-H "Authorization: Bearer <DATASOURCES:CREATE token>" \
-G -X POST "https://api.tinybird.co/v0/analyze" \
--data-urlencode "url=https://example.com/file"
```

ANALYZE RESPONSE

```
{
    "analysis": {
        "columns": [
```

```
            {
                "path": "$.a_nested_array.nested_array[:]",
                "recommended_type": "Array(Int16)",
                "present_pct": 3,
                "name": "a_nested_array_nested_array"
            },
            {
                "path": "$.an_array[:]",
                "recommended_type": "Array(Int16)",
                "present_pct": 3,
                "name": "an_array"
            },
            {
                "path": "$.field",
                "recommended_type": "String",
                "present_pct": 1,
                "name": "field"
            },
            {
                "path": "$.nested.nested_field",
                "recommended_type": "String",
                "present_pct": 1,
                "name": "nested_nested_field"
            }
        ],
        "schema": "a_nested_array_nested_array Array(Int16) `json:$.a_nested
    },
    "preview": {
        "meta": [
            {
                "name": "a_nested_array_nested_array",
                "type": "Array(Int16)"
            },
            {
                "name": "an_array",
                "type": "Array(Int16)"
            },
            {
                "name": "field",
                "type": "String"
            },
            {
                "name": "nested_nested_field",
```

```
                "type": "String"
            }
        ],
        "data": [
            {
                "a_nested_array_nested_array": [
                    1,
                    2,
                    3
                ],
                "an_array": [
                    1,
                    2,
                    3
                ],
                "field": "test",
                "nested_nested_field": "bla"
            }
        ],
        "rows": 1,
        "statistics": {
            "elapsed": 0.000310539,
            "rows_read": 2,
            "bytes_read": 142
        }
    }
}
```


The `columns` attribute contains the guessed columns and for each one:

- `path`: The JSONPath syntax in the case of NDJSON/Parquet files
- `recommended_type`: The guessed database type
- `present_pct`: If the value is lower than 1 then there was nulls in the sample used for guessing
- `name`: The recommended column name

The `schema` attribute is ready to be used in the Data Sources API

The `preview` contains up to 10 rows of the content of the file.

Spain
Calle del Dr. Fourquet, 27
28012 Madrid

USA
41 East 11th Street 11th floor
New York, NY 10003