USPTO Patent Data Processor

Overview

A Python command-line tool for downloading and processing USPTO (United States Patent and Trademark Office) patent data. Supports both single year and year range processing for patent grants and applications.

Features

- Download patent data from USPTO
- Support for both grant and application patents
- Single year or year range processing
- Automatic file unzipping
- Database storage for extracted examples and their classification (Prophetic or Non-Prophetic)

Requirements

- Python 3.6+
- Internet connection
- Sufficient disk space for patent data

Usage

Command Format

Single year:

```
python scripts/analyse.py --year 2015 --kind grant --path data
```

Year range:

```
python scripts/analyse.py --year-range 2015 2017 --kind grant --path data
```

Arguments

Argument	Description	Required	Default	Valid Values
`--year`	Single year to process	Yes*	None	1976-2025
`--year-range`	Start and end years	Yes*	None	1976-2025
`--kind`	Patent document type	No	grant	grant, application
`--path`	Output directory	No	data	Valid path

* Either `--year` or `--year-range` must be specified

Examples

Download grant patents for 2015:

```
python scripts/analyse.py --year 2015 --kind grant --path data
```

Download applications from 2015-2017:

```
python scripts/analyse.py --year-range 2015 2017 --kind application --path
patent_data
```

Output Structure

```
data/
├── patent_grants_2015/
│    ├── downloaded_patents.xml
│    └── ...
├── patent_grants_2016/
└── patent_grants_2017/
```

Validation Rules

- Years must be between 1976-2025
- When using year range, start year must be ≤ end year
- Patent kind must be either "grant" or "application"
- Output path must be valid or creatable

Error Handling

- Invalid years trigger validation error
- Invalid patent types trigger validation error
- Invalid paths attempt creation or raise error
- Invalid year ranges show error message

Processing Steps

1. Validates input arguments
2. Downloads patent data from USPTO
3. Unzips downloaded files
4. Extracted and classifies examples
5. Stores examples and statistics in a database
6. Provides progress feedback

Notes

- Downloads occur automatically
- Unzipping happens immediately after download
- Progress messages show current status
- Existing files are preserved