THE UPTRENZ PLATFORM

Multi-domain support [1]:

- **System-level**: Data belongs to distinct system and is stored separately
- **Item-level**: Domains have different data types (e.g., items) which may share some attributes

**Microservices-based system architecture.** Provides a layer on top of the ScaR [2] framework to dynamically configure an application domain and to instantly provide an API to:

- Upload data about **items**
- Upload data about **users**
- Upload the corresponding **interaction** data
- **Request recommendations**

**Domain-specific data model.** The underlying algorithm needs to unambiguously know which source of information should be used to calculate the recommendations. The Uptrendz platform allows generating customized data upload APIs for multiple item and user entities with the following data types:

<table>
<thead>
<tr>
<th>Type</th>
<th>Sub-Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Categorical</td>
<td>Single</td>
<td>String value, which usually represents a category. Used for post-filtering recommendation results.</td>
</tr>
<tr>
<td>Text</td>
<td>Multiple</td>
<td>List of string values, which usually represent an array of categories. Used for post-filtering recommendation results.</td>
</tr>
<tr>
<td>Free Text</td>
<td>English</td>
<td>English text, which is processed and utilized for content-based recommendations.</td>
</tr>
<tr>
<td></td>
<td>German</td>
<td>German text, which is processed and utilized for content-based recommendations.</td>
</tr>
<tr>
<td>Numeric</td>
<td>Integer</td>
<td>Used for post-filtering recommendations (e.g., user age).</td>
</tr>
<tr>
<td></td>
<td>Real</td>
<td>Used for post-filtering recommendations (e.g., price).</td>
</tr>
<tr>
<td>Date</td>
<td>-</td>
<td>Date information for the respective entity (e.g., creation date)</td>
</tr>
</tbody>
</table>

The customized interaction API can support registered users, anonymous sessions or both. It also supports defining an unrestricted amount of explicit and implicit interaction types.

DEMO WALKTHROUGH: SYSTEM LEVEL DOMAIN

**Goal:** Create a real-time movie recommender and use it in an API-centric for an online setting.

**Code:** [https://github.com/lacic/ECIR2023Demo](https://github.com/lacic/ECIR2023Demo)

**API Setup:**
- Create a movie domain on the system-level
- Configure what meta-data is available for movies
- Configure what meta-data is available for users
- Configure that we are using explicit rating interactions
- Customize recommendation scenarios with appropriate algorithms and post-filtering logic

ITEM-LEVEL DOMAIN

**Entrepreneurial startup founding.** Need to define recommendable entities which users can interact with on the Cogsteps platform:

- **D1**: Co-founder candidates that should be matched with a startup-idea.
- **D2**: Innovations to be discovered by incubators, innovation hubs, and accelerators.
- **D3**: Educational materials until the innovation idea matures enough to form a start-up.

**Recommender customization.** Defining a recommendation scenario involves configuring a specific algorithm on the data of a respective item-level domain (e.g., innovations).

REFERENCES
