

ADAMS

Advanced **D**ata mining **A**nd **M**achine learning **S**ystem

Module: adams-ufdl-core



Peter Reutemann

July 17, 2024

©2019-2023



THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato



Except where otherwise noted, this work is licensed under
<http://creativecommons.org/licenses/by-sa/4.0/>

Contents

1	UFDL	5
2	Flow	7
2.1	Actions	7
2.2	EnterManyValues extensions	10
2.3	Conversions	10
	Bibliography	13

Chapter 1

UFDL

UFDL, or user-friendly deep-learning, enables users with little or no deep-learning expertise to maintain datasets for deep-learning tasks such as image classification and object detection and also build deep-learning models using these.

Chapter 2

Flow

Using the flow *adams-ufdl-core-manage-backend.flow*, you can manage users, teams, projects and licenses.

The following standalones can be used:

- *UFDLConnection* – defines the host and login credentials for connecting to a UFDL API backend.

The following sources can be used:

- *UFDLSource* – uses the specified UFDL source action to output data.

The following transformers can be used:

- *UFDLCreateJob* – generates a job (interactively) from a job template.
- *UFDLExtractAndTransfer* – extracts the primary key (PK) or description of a list item in a variable and transfers it into another variable.
- *UFDLGetPK* – retrieves the primary key from a UFDL passing through.
- *UFDLTransformer* – uses the specified UFDL transform action to process data.

The following sinks can be used:

- *UFDLSink* – uses the specified UFDL sink action to consume data.

2.1 Actions

The following source actions can be used:

- *CreateCudaVersion* – creates a CUDA version.
- *CreateDockerImage* – creates a docker image.
- *CreateFramework* – creates a framework.
- *CreateHardwareGeneration* – creates a hardware generation.
- *CreateJobType* – creates a job type.
- *CreateLogEntry* – creates a log entry.
- *CreateLicense* – creates a license.
- *CreatePretrainedModel* – creates a pretrained model.

- *CreateProject* – creates a project.
- *CreateTeam* – creates a team.
- *CreateUser* – creates a user.
- *CurrentUser* – forwards the user logged in via the current context.
- *Generic* – for making generic API calls.
- *ListCudaVersions* – lists all the cuda versions.
- *ListDatasets* – lists all the datasets.
- *ListDockerImages* – lists all the docker images.
- *ListDomains* – lists all the domains.
- *ListFrameworks* – lists all the frameworks.
- *ListHardwareGenerations* – lists all the hardware generations.
- *ListJobs* – lists all the jobs.
- *ListJobOutputs* – lists all the outputs from a specific job.
- *ListJobTemplates* – lists all the job templates.
- *ListJobTypes* – lists all the job types.
- *ListLicenses* – lists all the licenses.
- *ListLogEntries* – lists all the log entries.
- *ListNodes* – lists all the nodes.
- *ListProjects* – lists all the teams.
- *ListTeams* – lists all the teams.
- *ListUsers* – lists all the users.
- *Null* – dummy, does nothing.

The following transformer actions can be used:

- *AbortJob* – aborts a job (via PK; admin only - removes output and basically restarts it).
- *AddJobOutput* – allows uploading of a job output.
- *AddTeamMember* – adds team member(s) via PK to the team passing through.
- *CancelJob* – cancels a job (via PK; admin only).
- *ClearDataset* – removes meta-data and annotations from a dataset (via PK or name).
- *CopyDataset* – creates a copy of a dataset (via PK or name).
- *DeleteCudaVersion* – deletes a CUDA version (via PK or name).
- *DeleteDataset* – deletes a dataset (via PK or name).
- *DeleteDockerImage* – deletes a docker image (via PK).
- *DeleteFramework* – deletes a framework (via PK).
- *DeleteHardwareGeneration* – deletes a hardware generation (via PK or name).
- *DeleteJob* – deletes a job (via PK).
- *DeleteJobOutput* – deletes a job output (via name).
- *DeleteJobTemplate* – deletes a job template (via PK).
- *DeleteJobType* – deletes a job type (via PK or name).
- *DeleteLicense* – deletes a license (via PK).
- *DeleteLogEntry* – deletes a log entry (via PK).
- *DeleteNode* – deletes a node (via PK).
- *DeletePretrainedModel* – deletes a pretrained model (via PK or name).

- *DeleteProject* – deletes a project (via PK or name).
- *DeleteTeam* – deletes a team (via PK or name).
- *DeleteUser* – deletes a user (via PK or name).
- *ExportJobTemplate* – exports a Job template as JSON.
- *ImportJobTemplate* – imports a Job template from JSON.
- *LoadCudaVersion* – loads a CUDA version (via PK or version).
- *LoadDataset* – loads a dataset (via PK or name).
- *LoadDockerImage* – loads a docker image (via PK).
- *LoadFramework* – loads a framework (via PK).
- *LoadHardwareGeneration* – loads a hardware generation (via PK or name).
- *LoadJob* – loads a job (via PK).
- *LoadJobTemplate* – loads a job template (via PK or name).
- *LoadLicense* – loads a license (via PK or name).
- *LoadNode* – loads a node (via PK or IP).
- *LoadPretrainedModel* – loads a pretrained model (via PK or name).
- *LoadProject* – loads a project (via PK or name).
- *LoadTeam* – loads a team (via PK or name).
- *LoadUser* – loads a user (via PK or name).
- *MergeDatasets* – for merging two datasets.
- *PassThrough* – dummy, does nothing.
- *RemoveTeamMember* – removes member(s) from the team passing through.
- *ReinstateDataset* – reinstates ("undeletes") a dataset.
- *ReinstateJob* – reinstates ("undeletes") a job.
- *ReinstateProject* – reinstates ("undeletes") a project.
- *ReinstateTeam* – reinstates ("undeletes") a team.
- *Reinstateuser* – reinstates ("undeletes") a user.
- *ReleaseJob* – releases a job (via PK; voluntarily done by node).
- *ResetJob* – resets a job for re-execution.
- *UpdateDocker* – updates a docker image.
- *UpdateLicense* – updates a license.
- *UpdateLicenseSubDescriptors* – updates the permissions, conditions and limitations of a license
- *UpdatePassword* – updates the password of a user.
- *UpdatePretrainedModel* – updates a pre-trained model.
- *UpdateProject* – updates a project.
- *UpdateTeam* – updates a team.
- *UpdateTeamMember* – updates team member(s).
- *UpdateUser* – updates a user (excluding password).
- *UploadPretrainedModelFile* – uploads a pretrained model binary for a pre-trained model.

The following sink actions can be used:

- *DownloadDataset* – for downloading a dataset.
- *DownloadJobOutput* – for downloading a job output.
- *DownloadPretrainedModel* – for downloading a pretrained model.
- *Null* – dummy, does nothing.

2.2 EnterManyValues extensions

The following extensions of value definitions used by the *EnterManyValues* source make it easier for users to select PKs:

- *UFDLCudaVersionList* – for selecting a CUDA version
- *UFDLCudaVersionChooser* – for selecting one or more CUDA versions
- *UFDLDataSetChooser* – for selecting one or more datasets
- *UFDLDataSetList* – for selecting a dataset
- *UFDLDockerImageChooser* – for selecting one or more docker images
- *UFDLDockerImageList* – for selecting a docker image
- *UFDLDomainChooser* – for selecting one or more domains
- *UFDLDomainList* – for selecting a domain
- *UFDLFrameworkChooser* – for selecting one or more frameworks
- *UFDLFrameworkList* – for selecting a framework
- *UFDLHardwareGenerationChooser* – for selecting one or more hardware generations
- *UFDLHardwareGenerationList* – for selecting a hardware generation
- *UFDLJobChooser* – for selecting one or more jobs
- *UFDLJobList* – for selecting a job
- *UFDLJobTemplateChooser* – for selecting one or more job templates
- *UFDLJobTemplateList* – for selecting a job template
- *UFDLJobTypeChooser* – for selecting one or more job types
- *UFDLJobTypeList* – for selecting a job type
- *UFDLLicenseChooser* – for selecting one or more licenses
- *UFDLLicenseFilter* – for selecting a license via a filter dialog
- *UFDLLicenseList* – for selecting a license
- *UFDLNodeChooser* – for selecting one or more nodes
- *UFDLNodeList* – for selecting a node
- *UFDLPretrainedModelChooser* – for selecting one or more pretrained models
- *UFDLPretrainedModelList* – for selecting a pretrained model
- *UFDLProjectChooser* – for selecting one or more projects
- *UFDLProjectList* – for selecting a project
- *UFDLTeamChooser* – for selecting one or more teams
- *UFDLTeamList* – for selecting a team
- *UFDLUserChooser* – for selecting one or more users
- *UFDLUserList* – for selecting a user

2.3 Conversions

The following conversions are available:

- *UFDLCudaVersionToSpreadSheet* – converts the UFDL cuda version to a spreadsheet.
- *UFDLDataSetToListItem* – converts a UFDL Dataset to a list item string.
- *UFDLDataSetToSpreadSheet* – converts the UFDL dataset to a spreadsheet.

- *UFDLDockerImageToSpreadSheet* – converts the UFDL docker image to a spreadsheet.
- *UFDLExtractFileNameFromFile* – converts a file name into an name used in datasets
- *UFDLExtractListItemPK* – extracts the PK from a list item output by *EnterManyValues*.
- *UFDLFrameworkToSpreadSheet* – converts the UFDL framework to a spreadsheet.
- *UFDLHardwareGenerationToSpreadSheet* – converts the UFDL hardware generation to a spreadsheet.
- *UFDLJobToSpreadSheet* – converts the UFDL job to a spreadsheet.
- *UFDLJobOutputToSpreadSheet* – converts the UFDL job output to a spreadsheet.
- *UFDLJobTemplateToSpreadSheet* – converts the UFDL job template to a spreadsheet.
- *UFDLJobTypeToSpreadSheet* – converts the UFDL job type to a spreadsheet.
- *UFDLJsonObjectToString* – converts a UFDL object that wraps a JSON object into a JSON string.
- *UFDLLicenseToListItem* – converts a UFDL License to a list item string.
- *UFDLLogEntryToSpreadSheet* – converts a UFDL log entry into a spreadsheet.
- *UFDLNodeToSpreadSheet* – converts a UFDL node into a spreadsheet.
- *UFDLPretrainedToSpreadSheet* – converts a UFDL pretrained model into a spreadsheet.
- *UFDLProjectToListItem* – converts a UFDL Project to a list item string.
- *UFDLProjectToSpreadSheet* – converts the UFDL project information to a spreadsheet.
- *UFDLTeamToListItem* – converts a UFDL Team to a list item string.
- *UFDLTeamToSpreadSheet* – converts the UFDL team information to a spreadsheet.
- *UFDLUserToListItem* – converts a UFDL User to a list item string.
- *UFDLUserToSpreadSheet* – converts the UFDL user information to a spreadsheet.

Bibliography

- [1] *ADAMS* – Advanced Data mining and Machine learning System
<https://adams.cms.waikato.ac.nz/>