

Quevedo: Annotation and Processing of Graphical Languages



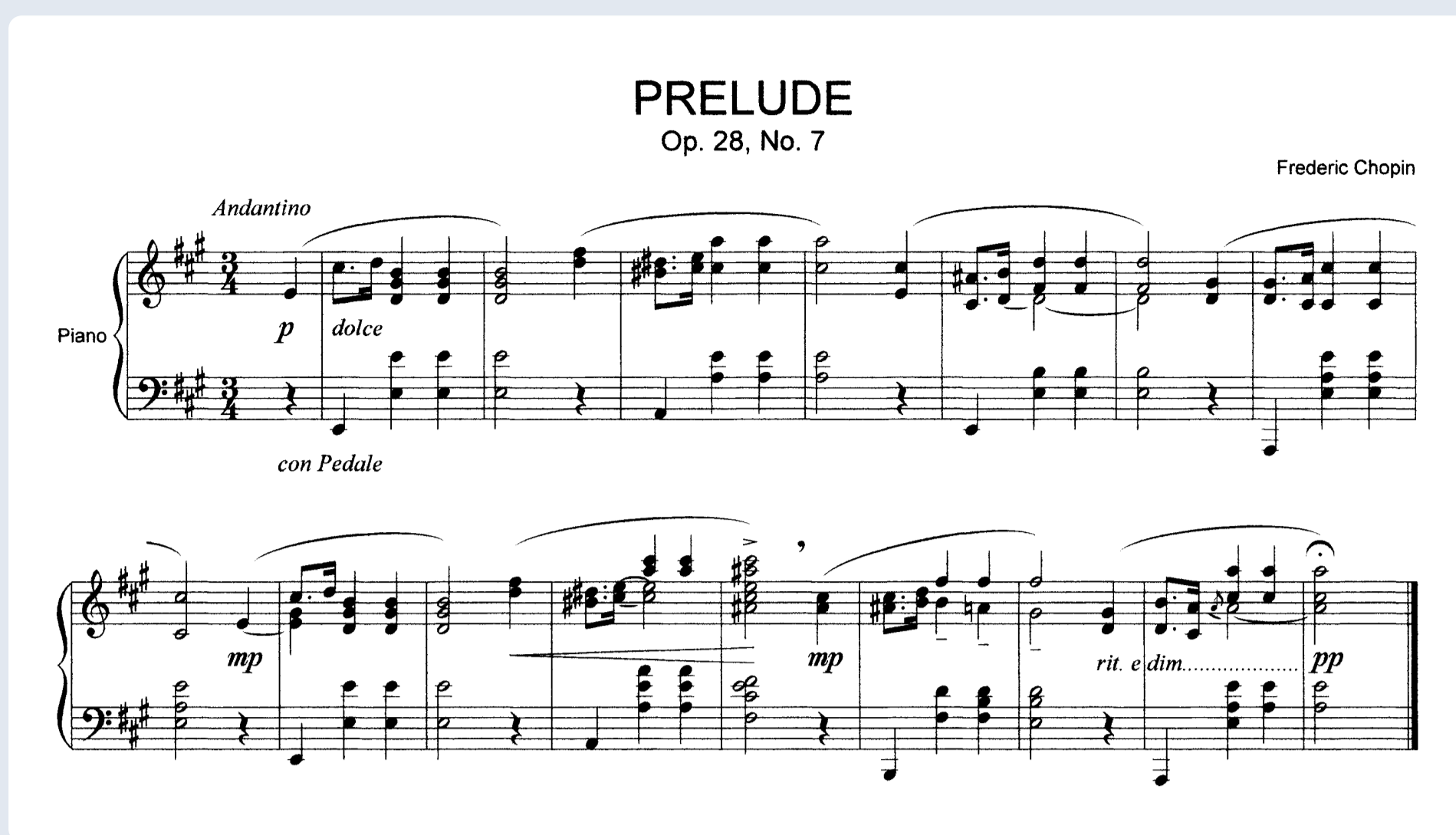
Antonio F. G. Sevilla
afgs@ucm.es

Alberto Díaz Esteban
albertodiaz@fdi.ucm.es

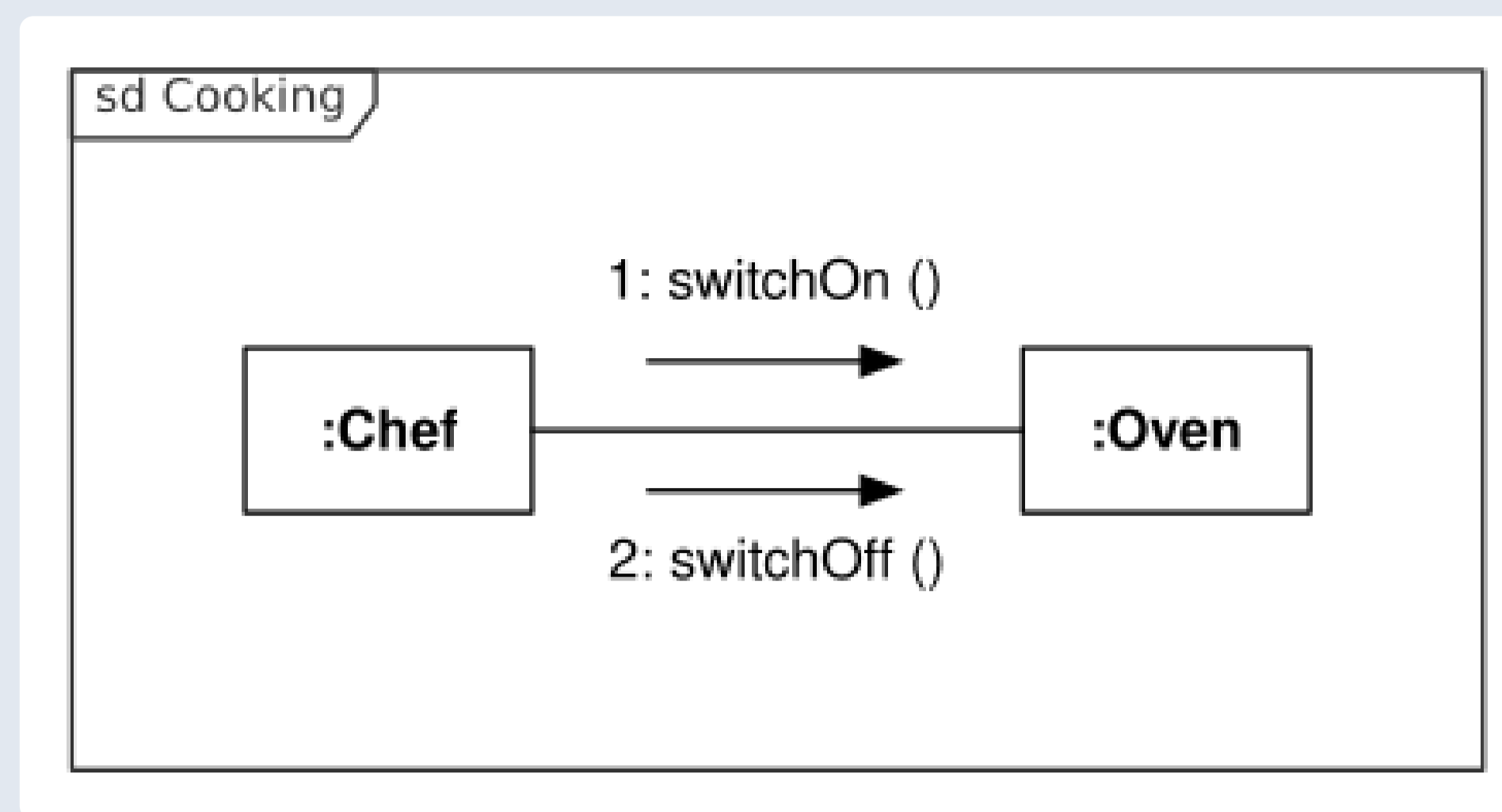
José María Lahoz-Bengoechea
jmlahoz@ucm.es



Universidad Complutense de Madrid



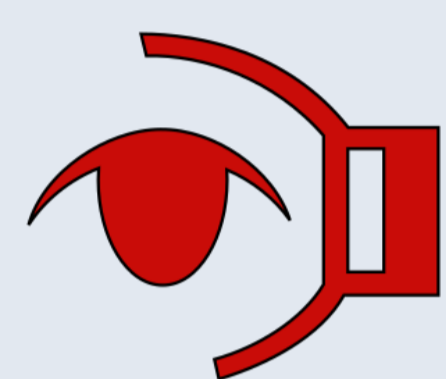
Graphical languages, such as musical notation, use the **2D** arrangement of symbols to **convey meaning**, using convention, abstraction and the fundamental sign-signified relationship of language.



The **relative location** of symbols is meaningful, such as in this UML diagram example, as well as their shape or direction, which can alter meaning.



SignWriting is a graphical language used to transcribe the gestures and movements of **Sign Languages**, using the 2D page to capture the multimodal 3D reality of signing.



Quevedo

Quevedo is a python **library** and command line **application** for creating, annotating and managing datasets of graphical languages, with a focus on the training and evaluation of machine learning algorithms for their recognition.

Features

- **Dataset management**, including hierarchical dataset organization, subset partitioning, and semantically guided data augmentation.
- Structural and visual **annotation** of source images using a **web interface**, with support for different users and the live visualization of data processing scripts.
- **Deep learning** network management, training, configuration and evaluation, using darknet.

Install & Use

- Quevedo requires `python >= 3.7`, and can be installed from PyPI:

```
$ pip install quevedo[web]
```

- Create a dataset, visually annotate it using your own browser, and train and evaluate neural networks:

```
$ quevedo -D new/dataset create ; cd new/dataset  
$ quevedo web  
$ quevedo -N network_name prepare train test
```

- More documentation available at <https://agarsev.github.io/quevedo>.

Deep Learning

Machine learning techniques developed in the field of **computer vision** are necessary to adequately process graphical languages. While the researcher can use any toolkit and algorithm they prefer, Quevedo includes a module to facilitate the use of deep learning **neural networks** with Quevedo datasets.

Pipelines

Quevedo allows you to train different neural networks to recognize different objects and features. These networks can then be composed into a pipeline to build an **expert system**, capable of performing a bigger task than each of the networks by themselves.

For example, a detection network can first locate the graphemes within a logogram, and then specialist networks be used to classify each of the graphemes.



Open Software License 3.0



github.com/agarsev/quevedo



ucm.es/visse



VisSE » logograms/A1_T1 » 91

Metadata meaning: Coffee
notes: T000: Check the direction of movement with informants

Graphemes

CLASS	SHAPE	VAR	ROT	REF
HAND	picam-	hh	N	y
HAND	pir+O	hb	NW	n
ARRO	b	VAR	W	REF
ARRO	b	VAR	W	REF
ARC	sf	VAR	N	REF

To automatically process graphical languages, complex, visual annotation is needed. This annotation includes information for the whole **logogram**, as well as locative data for the different **graphemes**. Each grapheme is also given a set of tags to identify its meaning and linguistic features.

We have used Quevedo to create the **VisSE** annotated corpus of Spanish SignWriting: <https://zenodo.org/record/6337885>.



Toy Arithmetic » logograms/operations » 5

Metadata filename: subtraction

Graphemes

type	value
number	6
number	1
number	4
number	2
number	1
number	4
number	5
number	3
number	4
number	6
number	8
number	9
symbol	-

Quevedo is **domain-agnostic**, meaning you can use it to process different graphical languages or similar visual problems. In this example, included with Quevedo's source, we have annotated the graphical language of elementary arithmetic. Bring your own annotation schema!