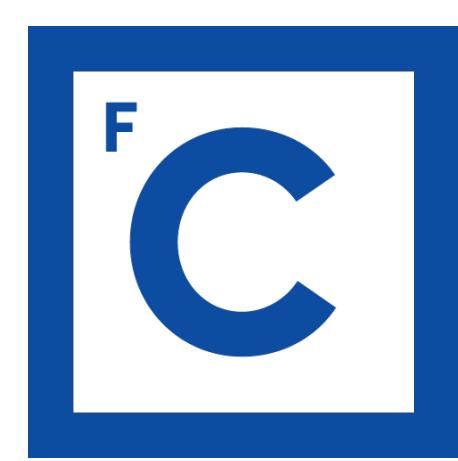


# Universal Grammatical Dependencies for Portuguese with CINTIL Data, LX Processing and CLARIN Support



Ciências ULisboa

António Branco, João Silva, Luís Gomes and João Rodrigues  
Faculty of Sciences of the University of Lisbon

LREC 2022 Marseille

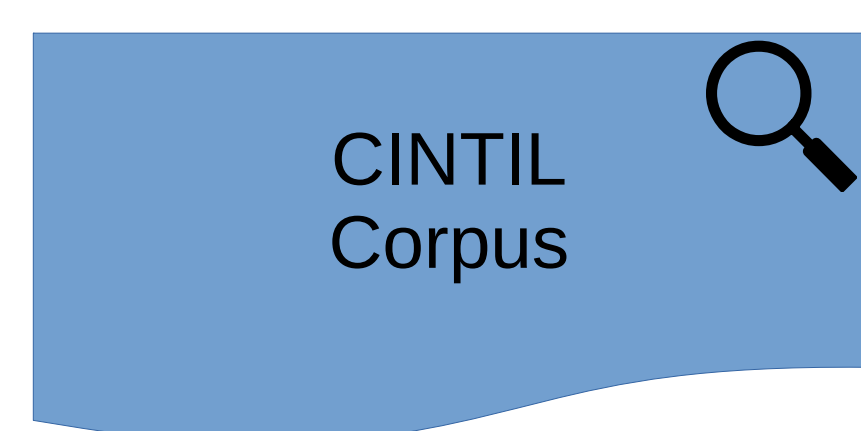
The largest UD POS corpus and UD treebank for Portuguese, and their corresponding POS tagger and dependency parser

Manually verified annotated corpora

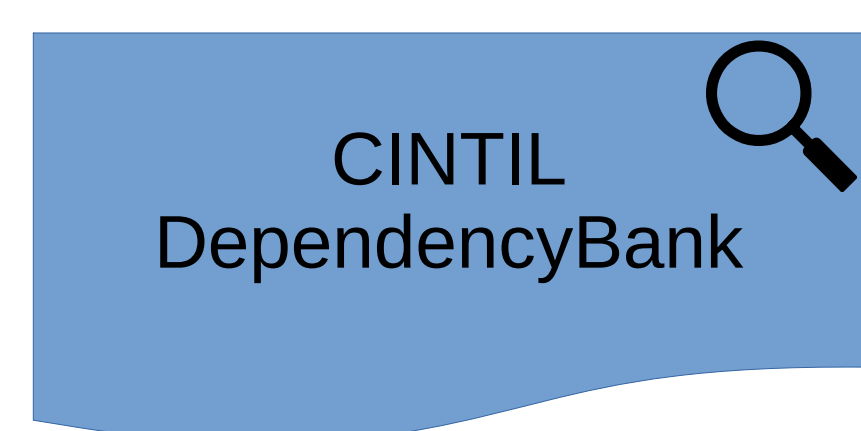
Automatic conversion to Universal Dependencies

UD data sets and their corresponding tools

All resources freely available in the PORTULAN CLARIN Repository  
<https://portulanclarin.net/repository/search/>



CINTIL Corpus



CINTIL DependencyBank

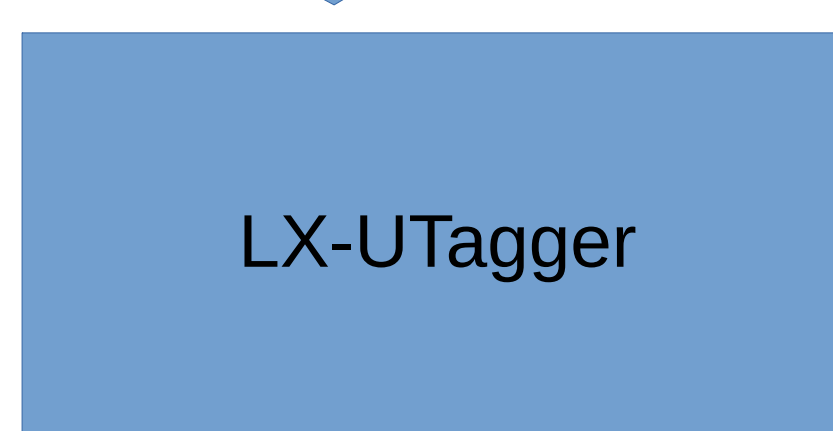


CINTIL DependencyBank PREMIUM



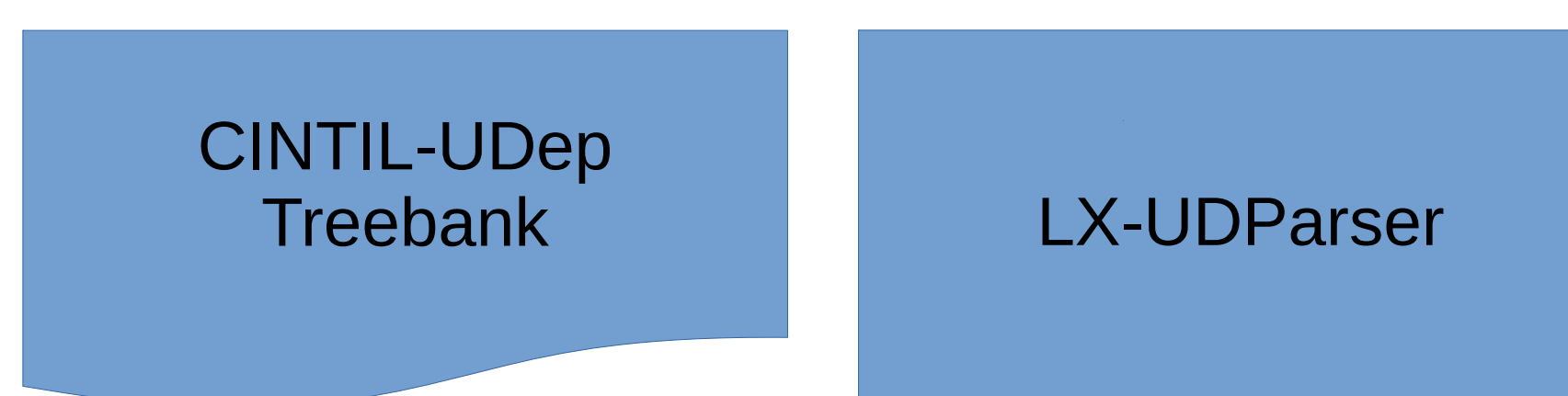
CINTIL-UPos

675k tokens



LX-UTagger

99.01% accuracy



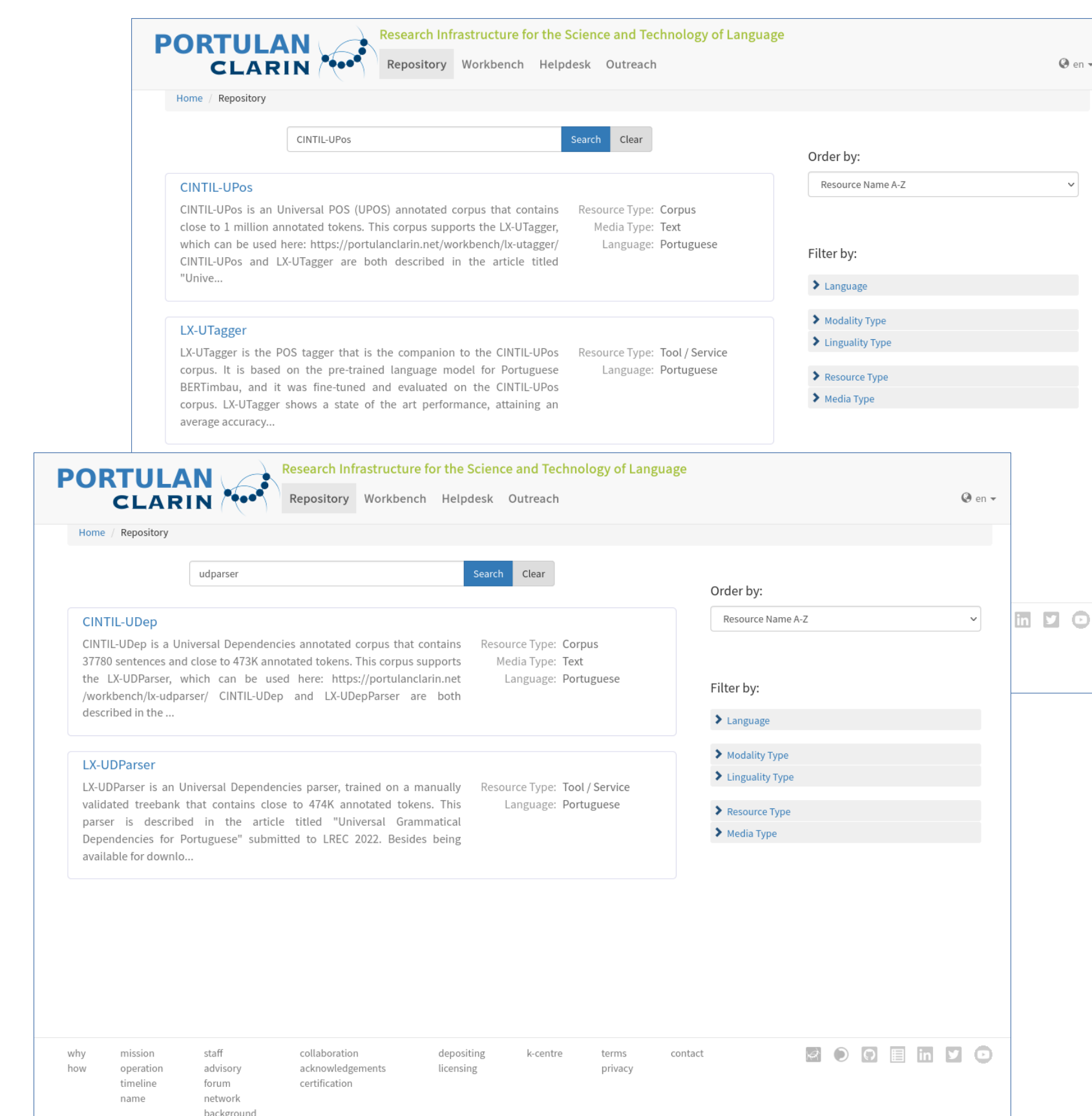
CINTIL-UDep Treebank

38k sentences



LX-UDParser

90.87 UAS / 88.01 LAS



Tools also available as online, web, batch, and notebook services  
<https://portulanclarin.net/workbench/lx-udparser/>

PORTULAN CLARIN Research Infrastructure for the Science and Technology of Language

Repository Workbench Helpdesk Outreach

Home / Workbench / LX-UDParser

**LX-UDParser**

Examples File Processing Notebook Web Service Documentation

Começaram bem os argentinos, desenvolvendo lances rápidos de contra-ataque.

Visualization format:  friendly  CONLL

Clear Parse

root -Root- Começaram bem os argentinos desenvolvendo lances rápidos de contra-ataque PUNCT

VERB ADV DET NOUN VERB NOUN ADJ ADP NOUN PUNCT

COMEÇAR BEM OS ARGENTINOS DESENVOLVER LANCE RÁPIDO CONTRA-ATAQUE

why mission staff collaboration depositing k-centre terms contact  
how operation advisory forum acknowledgements licensing privacy

Instructions to use this web service

The web service for this application is available at <https://portulanclarin.net/workbench/lx-udparser/>.

Below you find an example of how to use the web service with Python 3.

This example results in the response package. To install this package, use the command in the terminal from `pip3 install requests`.

To use this web service, you need an access key you can obtain by clicking in the button below. A key is valid for 31 days. It allows to submit a total of 500 million characters by means of requests with no more than 2000 characters each. It allows to enter 100,000 requests, at a rate of no more than 200 requests per hour.

For other enquiries, you should contact the helpdesk.

The input data and the respective output will be automatically deleted from our computer after being processed. No copies will be retained after your use of this service.

import json  
import requests  
# To install this library, enter in your terminal:  
# pip3 install requests

# This is a simple example to illustrate how you can use the LX-UDParser web service

# Requires: key as a string with your access key  
# Requires: text as a string with your text, with a maximum of 2000 characters, Portuguese text, with the input to be processed  
# Requires: request as a string, indicating the input to be used in the output, which can be either "CINTIL" or "UD" (universal dependencies)

# Requires: format as a string, indicating the output format, which can be either "friendly" or "CONLL"

# Getting access key status:

```
request_data = {  
    "method": "key_status",  
    "format": "UD",  
    "key": "key",  
}
```

```
request = requests.post(url, json=request_data)  
response_data = request.json()  
if "error" in response_data:  
    print("Error:", response_data["error"])  
else:  
    print("Key status")  
    print(json.dumps(response_data["result"], indent=1))
```

File processing

Input format: Input files must be in .txt FORMAT with UTF-8 ENCODING and contain PORTUGUESE TEXT. Input files and folders can also be compressed to the .zip format.

Drag here  
or  
Click here to select from your computer  
either a file or a compressed folder with files.

Privacy: The input file you upload and the respective output files will be automatically deleted from our computer after being processed and the result downloaded by you. No copies of your files will be retained after your use of this service.

Cancel Upload

This is only a preview of the notebook. To run, please choose one of the following options:

[Run on Binder](#) [Run on Google Colab](#) [Download from GitHub](#)

Using LX-UDParser to parse sentences and displaying dependency tree graphs

This is an example notebook that illustrates how you can use the LX-UDParser web service to parse sentences and how to visualize dependency tree graphs in a notebook.

Before you run this example, replace `ACCESS_KEY_NAME` by your web service access key, below:

```
!pip3 install requests  
!pip3 install requests
```

import requests

The next cell will take care of installing the `requests` and `lx-udparser` packages, if not already installed, and make them available to use in this notebook.

```
!pip3 install requests  
!pip3 install lx-udparser
```

def parse(text, request, format):  
 """  
 Args:  
 text: A string with a maximum of 2000 characters, Portuguese text, with the input to be processed.  
 request: either "CINTIL" or "UD" (universal dependencies).  
 format: either "friendly" or "CONLL".  
 Returns:  
 A string with the output according to specification in <https://portulanclarin.net/workbench/lx-udparser/>.  
 Raises:  
 A ValueError if an error occurs.  
 """  
 request\_data = {  
 "method": "parse",  
 "format": "UD",  
 "key": "key",  
 "text": text,  
 "request": request,  
 "format": format,  
 "key": LX-UDPARSER\_WS\_API\_KEY,  
 }  
 request = requests.post(LX-UDPARSER\_WS\_API\_URL, json=request\_data)  
 response\_data = request.json()  
 if "error" in response\_data:  
 raise ValueError(response\_data["error"])  
 else:  
 return response\_data["result"]

Let us test the function we just defined:

```
text = "Começaram bem os argentinos desenvolvendo lances rápidos de contra-ataque."  
format = "friendly"  
!pip3 install requests  
!pip3 install lx-udparser  
result = parse(text, request="CINTIL", format="friendly")  
print(result)
```

```
!pip3 install requests  
!pip3 install lx-udparser  
result = parse(text, request="CINTIL", format="CONLL")  
print(result)
```

id	form	lemma	upos	pos	feats	head	deprel	phrasal	phrasal
1	ESTÁ	DESENVOLVER	VERB	SP	{}	2	SP		SP
2	FRASE	FRASE	CONJ	FS	{}	1	FS		FS
3	servir	SERVI	VERB	FS	{}	0	ROOT		ROOT
4	para	PARA	PREP	FS	{}	3	FS		FS
5	teclar	TECLAR	VERB	FS	{}	0	ROOT		ROOT
6	o	O	DET	FS	{}	5	FS		FS
7	funcionamento	FUNCIIONAMENTO	NOUN	FS	{}	6	FS		FS
8	de	DE	PREP	FS	{}	7	FS		FS
9	parar	PARAR	VERB	FS	{}	0	ROOT		ROOT
10	dependencia	DEPENDENCIA	NOUN	FS	{}	8	FS		FS
11	de	DE	PREP	FS	{}	10	FS		FS
12	dependencia	DEPENDENCIA	NOUN	FS	{}	11	FS		FS
13	de	DE	PREP	FS	{}	12	FS		FS
14	de	DE	PREP	FS	{}	13	FS		FS
15	de	DE	PREP	FS	{}	14	FS		FS



This work was supported by PORTULAN CLARIN - Research Infrastructure for the Science and Technology of Language funded by Lisboa 2020, Alentejo 2020 and FCT - Fundação para a Ciência e Tecnologia under the grant PINFRA/22117/2016