

**UAM - Universidad Autónoma de Madrid**

# **Modelos libres de traducción automática y localización**

Antoni Oliver (aoliverg@uoc.edu)

# Objetivos e índice

## Objetivos de la sesión:

- Conocer los modelos de traducción libres (OpusMT y NLLB)
- Conocer los objetivos y componentes del proyecto MTUOC
- Saber cómo poner en marcha estos motores
  - En terminal
  - En MTUOC-server
- Saber cómo se lleva a cabo la recuperación de etiquetas HTML/XML en motores de traducción
- Saber cómo se lleva a cabo un proceso de *fine-tuning* de modelos OpusMT
- Métricas de evaluación automática
- Aplicación de los modelos libres a un proyecto de localización
  - Con OmegaT
  - Con tika de Okapi

# Corpus paralelos

# Opus corpora

<https://opus.nlpl.eu/>

# Modelos de traducción automática neuronal libres

## OpusMT

<https://github.com/Helsinki-NLP/Opus-MT>

<https://huggingface.co/Helsinki-NLP/opus-mt-en-es>

## Ejecución desde un script

```
from transformers import MarianMTModel, MarianTokenizer

src_text = ["This is a simple translation test.", "And this is another sentence."]

model_name = "Helsinki-NLP/opus-mt-en-es"
tokenizer = MarianTokenizer.from_pretrained(model_name)
model = MarianMTModel.from_pretrained(model_name)
translated = model.generate(**tokenizer(src_text, return_tensors="pt",
padding=True))
res = [tokenizer.decode(t, skip_special_tokens=True) for t in translated]
print(res)
```



## NLLB - No Language Left Behind

<https://ai.meta.com/research/no-language-left-behind/>

<https://huggingface.co/facebook/nllb-200-distilled-600M>

## Ejecución desde un script

```
from transformers import AutoTokenizer, AutoModelForSeq2SeqLM, pipeline

tokenizer = AutoTokenizer.from_pretrained("facebook/nllb-200-distilled-600M")
model = AutoModelForSeq2SeqLM.from_pretrained("facebook/nllb-200-distilled-600M")

translator = pipeline('translation', model=model, tokenizer=tokenizer,
src_lang='eng_Latn', tgt_lang='spa_Latn', max_length = 200)
src_text = ["This is a simple translation test.", "And this is another sentence."]
res = translator(src_text)
print(res)
```

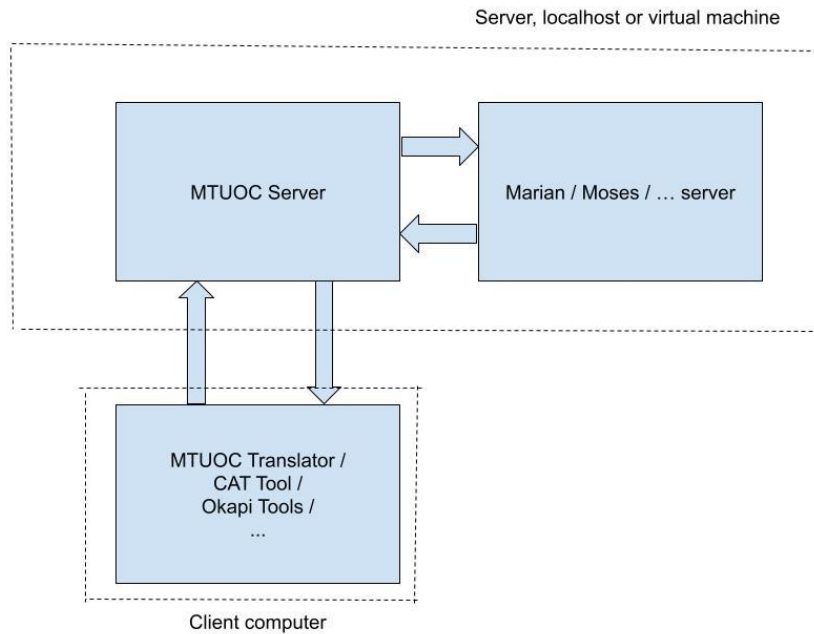
# El proyecto MTUOC

## Objetivos del proyecto

Facilitar la creación de corpus paralelos, el entrenamiento, evaluación e integración de motores de traducción automática neuronales (y estadísticos).

<https://mtuoc.github.io/>

# MTUOC-server



# OpusMT en MTUOC

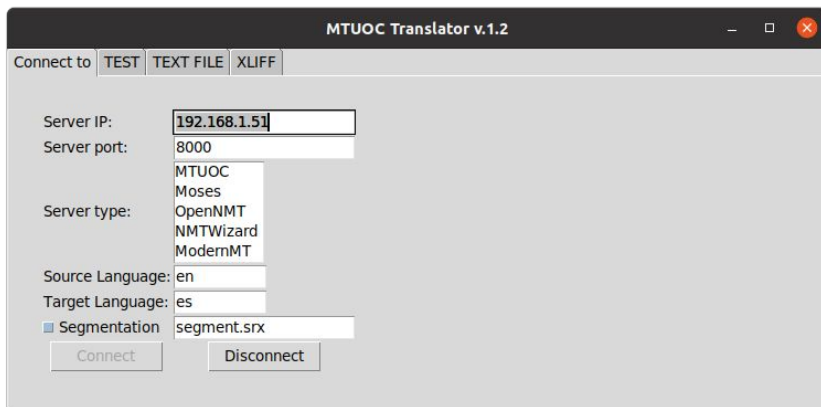
# config-server.yaml

```
MTEngine:
  MTengine: OpusMT
  #one of Marian, OpenNMT, Moses, GoogleTranslate, DeepL, Lucy, OpusMT, NLLB, Softcatalà, Apertium, Transformers, Aina
  SLcode: en
  TLcode: es
  multilingual: False
  #False or <tgtlang> or any multilingual code used by the system.

MTUOCServer:
  port: 8000
  type: MTUOC
  #one of MTUOC, Moses, ModernMT, OpenNMT, NMTWizard
  verbosity_level: 3
  log_file: log.log
  ONMT_url_root: "/translator"
  #specific configuration when acting as ONMT server
...
Transformers:
#use the same configuration for OpusMT
  model_path: ../opus-mt-en-es
  #model_path: Helsinki-NLP/opus-mt-tc-big-en-cat_oci_spa
  beam_size: 5
  num_hypotheses: 5
```

# Servidor en funcionamiento

```
2024-11-19 13:35:25.470616      3      MTUOC server started using MTUOC protocol
MTUOC server IP:      192.168.1.51
MTUOC server port:    8000
MTUOC server type:    MTUOC
```



MTUOC Translator v.1.2

Connect to: **TEST** | TEXT FILE | XLIFF

Server IP:

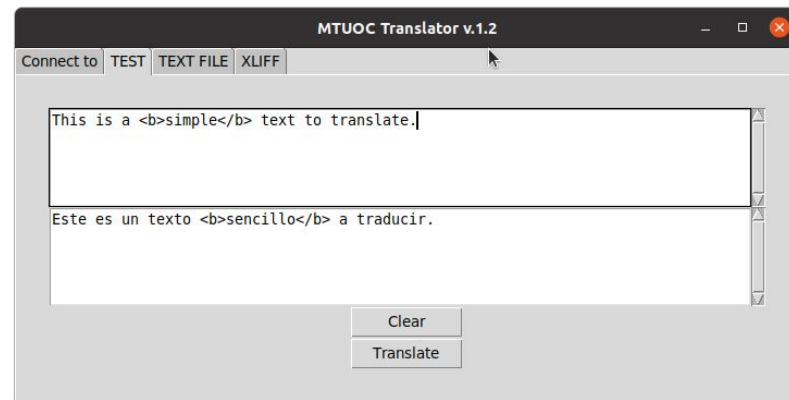
Server port:

Server type:  (dropdown menu open showing: MTUOC, Moses, OpenNMT, NMTWizard, ModernMT)

Source Language:

Target Language:

☒ Segmentation



MTUOC Translator v.1.2

Connect to: **TEST** | TEXT FILE | XLIFF



# Recuperación de etiquetas XML en TAN

[Tutorial: entrenamiento de modelos de alineación con fast\\_align para utilizarlos con MTUOC-server](#)

# Puesta en marcha de motores OpusMT en MTUOC-server

[Tutorial: poner en marcha motores OpusMT con MTUOC-server](#)

# Puesta en marcha del motores NLLB en MTUOC-server

[Tutorial: poner en marcha motores NLLB con MTUOC-server](#)

# Fine-tuning de modelos OpusMT

[Tutorial: Fine-tuning de modelos de OpusMT](#)

# OmegaT

Es necesario instalar el plugin:

<https://github.com/mtuoc/MTUOC-OmegaT-plugin>

[Tutorial: Uso de servidores MTUOC con OmegaT](#)

## Tikal de Okapi Tools

Okapi Tools: <https://okapiframework.org/>

Puede utilizar servidores ModernMT

Podemos poner en marcha el servidor MTUOC como ModernMT

```
./tikal.sh -t ../properties/Bundle.properties -sl en  
-tl es -seg segment.srx -mmt http://192.168.1.51:8000
```

## Conclusiones

- Existen modelos libres de traducción automática neuronal
- Estos modelos se pueden adaptar mediante el proceso de *fine tuning*
- Ventajas
  - Calidad comparable o mejor que los comerciales
  - Sin costes asociados sea cual sea el volumen
  - Confidencialidad (nada sale de nuestros servidores)
- Soberanía tecnológica

¡Muchas gracias por vuestra atención!

Antoni Oliver  
aoliverg@uoc.edu