Transliteration using Transformers

Alex de la Paz, Senior Data Scientist
AI & ML Ops Division, Data and Digital Innovation

4 January 2022

The overall classification of this presentation is:
UNCLASSIFIED

Classified by:
Derived from: N/A
Declassify on: N/A

NATIONAL GEOSPATIAL NGA INTELLIGENCE AGENCY

Purpose

- (U) Automate the transliteration process for speciality languages

Terminology

- (U) **NLP**: Natural language processing
- (U) **NLTK**: Natural Language Toolkit
- (U) **Source language**: A language passed as input to an algorithm.
- (U) **Target language**: A language returned as output of an algorithm.
- (U) **Diacritic**: A symbol that provides an alternative way of pronouncing a letter.
- (U) **Diacritization**: Replacing letters in a source language with their counterpart that contains a diacritic.
- (U) **Transliteration**: Converting the text in a source language into the equivalent characters in a target language, so that the source text may be read in the target language with proper pronunciation.
- (U) **Romanization**: Transliterating a source language text into the equivalent Latin characters.

Speciality languages with transliteration

- (U) Arab & Persian text without diacritics
- (U) Chinese text does not include whitespace
- (U) Automation of the transliteration process requires more than programmatic approach

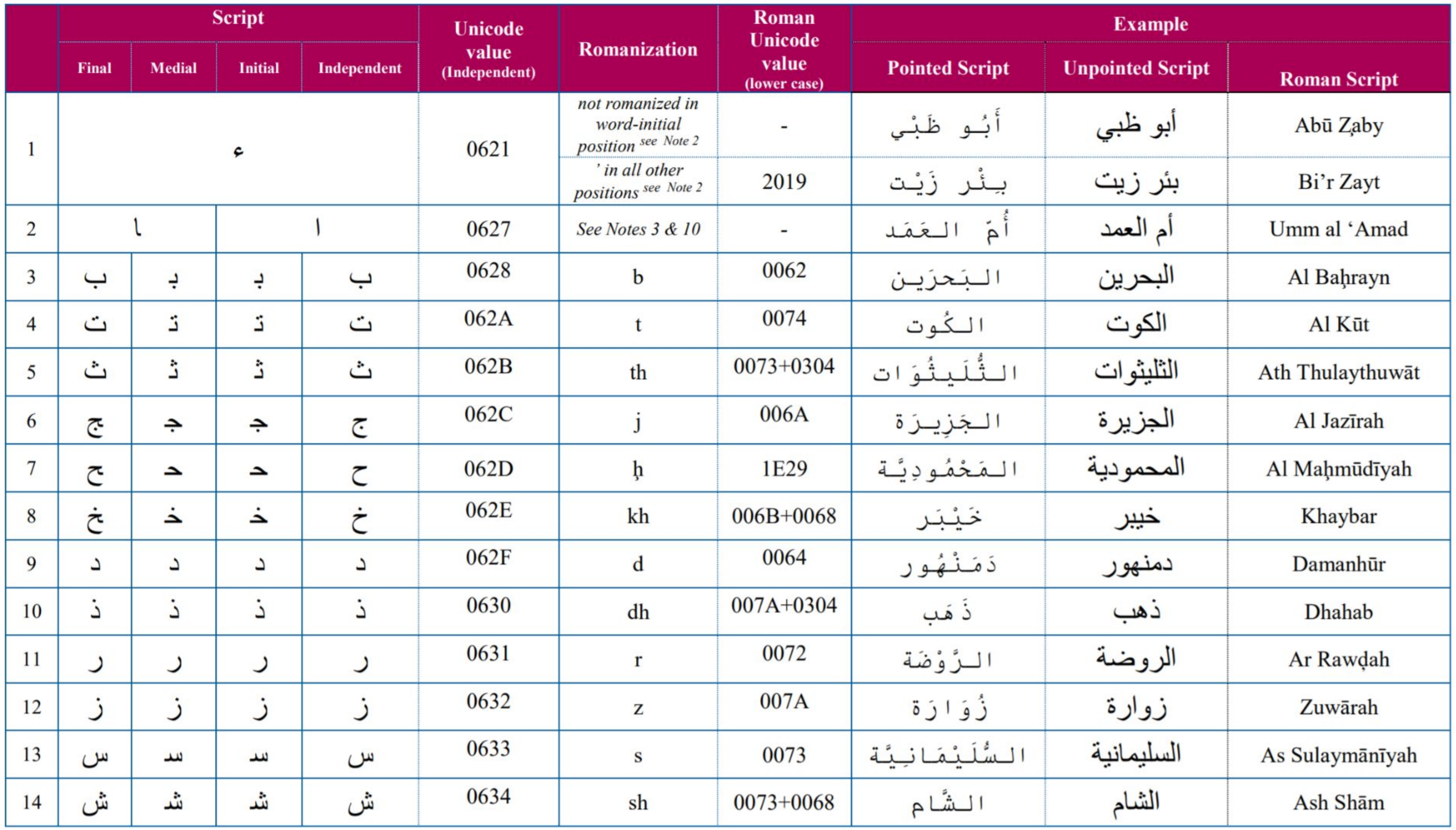
Romanization systems

- The **U.S. Board on Geographic Names (BGN)** and the **Permanent Committee on Geographical Names for British Official Use (PCGN)** jointly develop and/or approve romanization systems and Roman-script spelling conventions for the purpose of establishing standardized Roman-script spellings of those foreign geographical names that are written in non-Roman scripts or in Roman alphabets that contain special letters.

- Referred to as a **BGN/PCGN system**

BGN/PCGN 1956 System

Table 1: Standard Arabic Consonant Characters



	Script				Unicode value (Independent)	Romanization	Roman Unicode value (lower case)	Example		
	Final	Medial	Initial	Independent				Pointed Script	Unpointed Script	Roman Script
1	ء				0621	not romanized in word-initial position see Note 2	-	أَبُـو ظَبْـي	أبو ظبي	Abū Z̧aby
						’ in all other positions see Note 2	2019	بِـئْـر زَيْـت	بئر زيت	Bi’r Zayt
2	ـا		ا		0627	See Notes 3 & 10	-	أُمّ الـعَمَـد	أم العمد	Umm al ‘Amad
3	ـب	ـبـ	بـ	ب	0628	b	0062	الـبَحـرَيـن	البحرين	Al Baḩrayn
4	ـت	ـتـ	تـ	ت	062A	t	0074	الـكُـوت	الكوت	Al Kūt
5	ـث	ـثـ	ثـ	ث	062B	th	0073+0304	الـثُّـلَـيـثُـوَ ات	الثليثوات	Ath Thulaythuwāt
6	ـج	ـجـ	جـ	ج	062C	j	006A	الـجَزِيـرَة	الجزيرة	Al Jazīrah
7	ـح	ـحـ	حـ	ح	062D	ḩ	1E29	الـمَحْمُـودِيَّـة	المحمودية	Al Maḩmūdīyah
8	ـخ	ـخـ	خـ	خ	062E	kh	006B+0068	خَـيْـبَـر	خيبر	Khaybar
9	ـد	ـد	د	د	062F	d	0064	دَمَـنْـهُـور	دمنهور	Damanhūr
10	ـذ	ـذ	ذ	ذ	0630	dh	007A+0304	ذَ هَب	ذهب	Dhahab
11	ـر	ـر	ر	ر	0631	r	0072	الـرَّوْضَة	الروضة	Ar Rawḑah
12	ـز	ـز	ز	ز	0632	z	007A	زُوَ ا رَة	زوارة	Zuwārah
13	ـس	ـسـ	سـ	س	0633	s	0073	الـسُّلَـيْمَـا نِـيَّـة	السليمانية	As Sulaymānīyah
14	ـش	ـشـ	شـ	ش	0634	sh	0073+0068	الـشَّـام	الشام	Ash Shām

Language dataset pairs

Source text: Arabic	Target text: Latin
وادي	Wādī
خربة	Khirbat
قرية	Qaryah
مدرسة	Madrasah
تل	Tall
نهر	Nahr
مسجد	Masjid
جامع	Jāmi‘
مستشفى	Mustashfá
مستوصف	Mustawwşaf
مركز	Markaz
المعطن	Al Ma‘ţan
حدبة العجيري	Ḩadabat al ‘Ujayrī
⋮	⋮

Language dataset pairs

Shortest example

Source text: Arabic
Length: 2 characters

آل

Target text: Latin
Length: 2 characters

Āl

Longest example

Source text: Arabic
Length: 49 characters

مركز الأمومة و الطفولة و الولادة الطبيعية خورمكسر

Target text: Latin
Length: 67 characters

Markaz al Umūmah wa aţ Ţufūlah wa al Wilādah aţ Ţabī‘īah Khūrmaksar

Language dataset pairs

One-to-one mapping

Source text: Arabic
Length: 4 characters

وادي

Target text: Latin
Length: 4 characters

Wādī

ي	د	ا	و

W	ā	d	ī

Language dataset pairs

One-to-many mapping

Source text: Arabic
Length: 4 characters

خربة

Target text: Latin
Length: 7 characters

Khirbat

ة	ب	ر	خ

K	h	i	r	b	a	t

Language dataset pairs

Discontinuous character string

Source text: Arabic
Length: 8 characters

بيت محسن

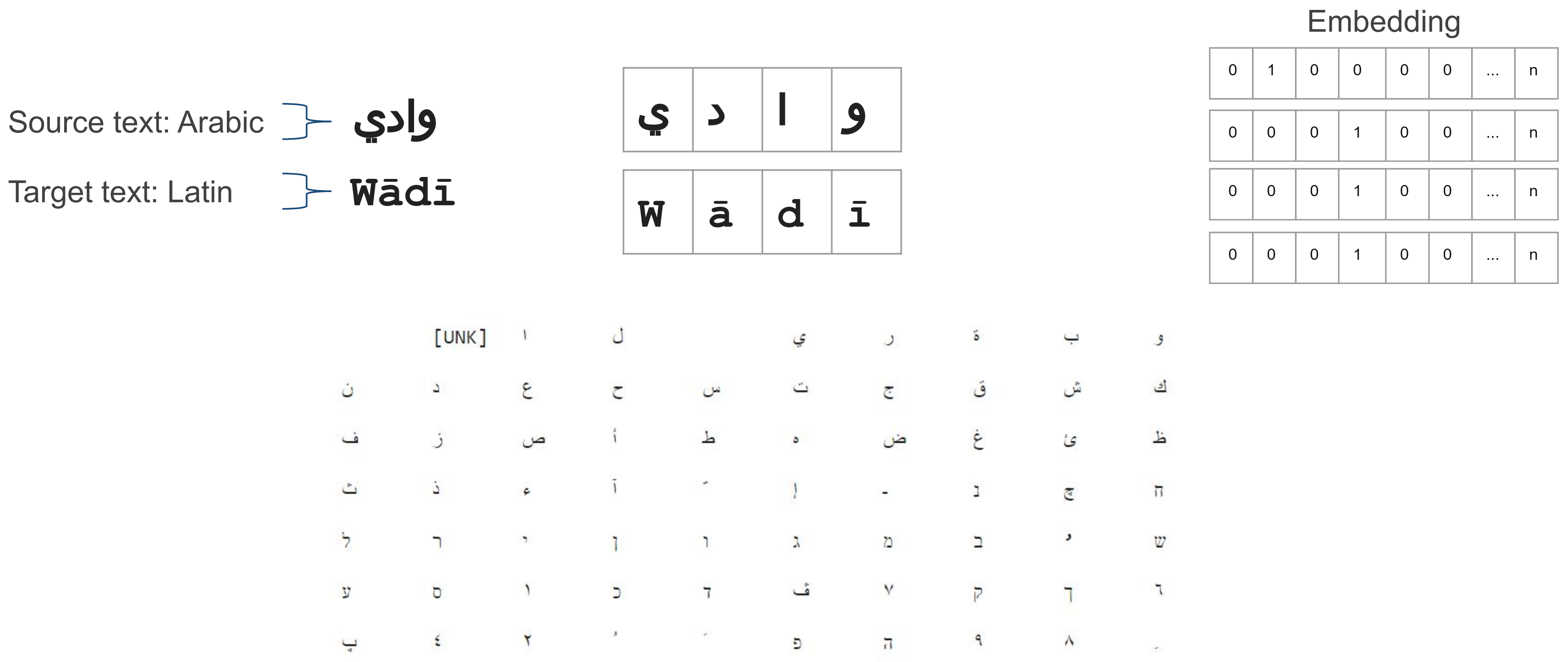
Target text: Latin
Length: 11 characters

Bayt Muḩsin

ن	س	ح	م	' '	ت	ي	ب

B	a	y	t	' '	M	u	ḩ	s	i	n

Character Embedding



Source text: Arabic } وادي

Target text: Latin } Wādī

ي	د	ا	و
W	ā	d	ī

Embedding

0	1	0	0	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n

	[UNK]	ا	ل		ي	ر	ة	ب	و
ن	د	ع	ح	س	ت	ج	ق	ش	ك
ف	ز	ص	أ	ط	ه	ض	غ	ئ	ظ
ث	ذ	ء	آ	ّ	إ	ـ	נ	چ	ח
ל	ר	י	ן	ו	ג	מ	ב	’	ש
ע	ס	١	כ	ד	ڤ	٧	ק	ך	٦
پ	٤	٢	،	ً	פ	ה	٩	٨	ـ

Character Embedding + Positional Embedding

Source text: Arabic } وادي

ي	د	ا	و

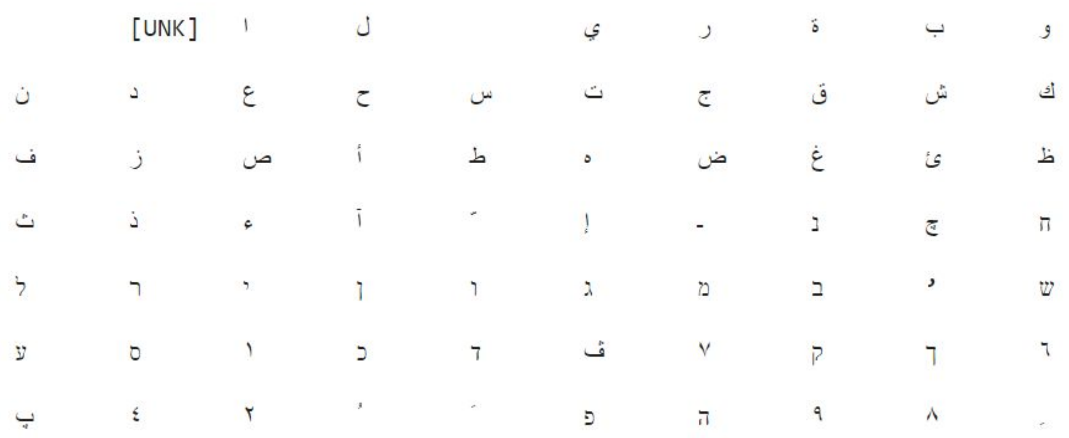
Embedding

0	1	0	0	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n

+

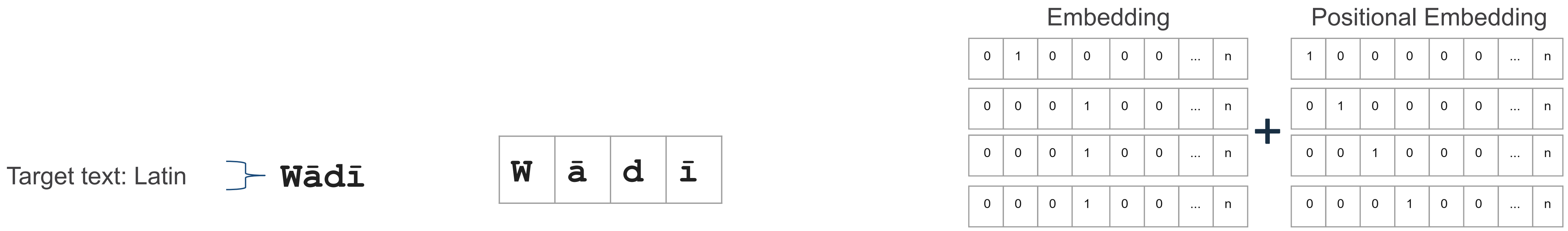
Positional Embedding

1	0	0	0	0	0	...	n
0	1	0	0	0	0	...	n
0	0	1	0	0	0	...	n
0	0	0	1	0	0	...	n



	[UNK]	ا	ل		ي	ر	ة	ب	و
ن	د	ع	ح	س	ت	ج	ق	ش	ك
ف	ز	ص	أ	ط	ه	ض	غ	ئ	ظ
ث	ذ	ء	آ	ً	إ	-	נ	چ	ח
ל	ר	ٔ	ן	ו	ג	מ	ב	،	ש
ע	ס	١	כ	ד	ڤ	٧	ק	ך	٦
پ	٤	٢	ُ	َ	פ	ה	٩	٨	ِ

Character Embedding + Positional Embedding



Target text: Latin

Wādī

W	ā	d	ī

Embedding

0	1	0	0	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n

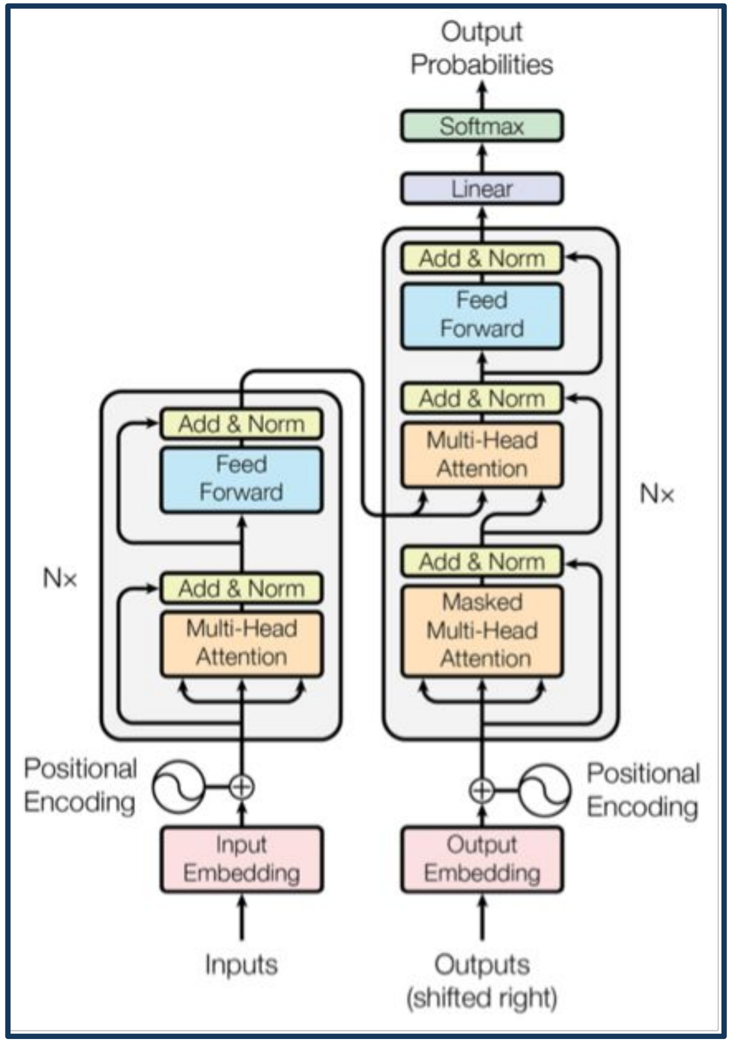
+

Positional Embedding

1	0	0	0	0	0	...	n
0	1	0	0	0	0	...	n
0	0	1	0	0	0	...	n
0	0	0	1	0	0	...	n

	[UNK]	a		]	[	h	l	r	ā
b	s	m	y	t	n	d	i	u	w
ʿ	j	ū	q	z	ḩ	f	Ḩ	ş	ţ
ḑ	g	ʾ	Ţ	,	á	Ā	Ḑ	Ū	Ī
ẖ	e	ṭ	v	ṣ	o	·	p	Ṭ	Ṣ
Ḏ	ḍ	_	ı						

Model training



Ashish Vaswani et al., “Attention is all you need” (2017)

Source text: Arabic

Target text: Latin

```
{آتزمون
,...
,...}
,...}
```

Embedding

0	1	0	0	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n

Positional Embedding

1	0	0	0	0	0	...	n
0	1	0	0	0	0	...	n
0	0	1	0	0	0	...	n
0	0	0	1	0	0	...	n

Embedding

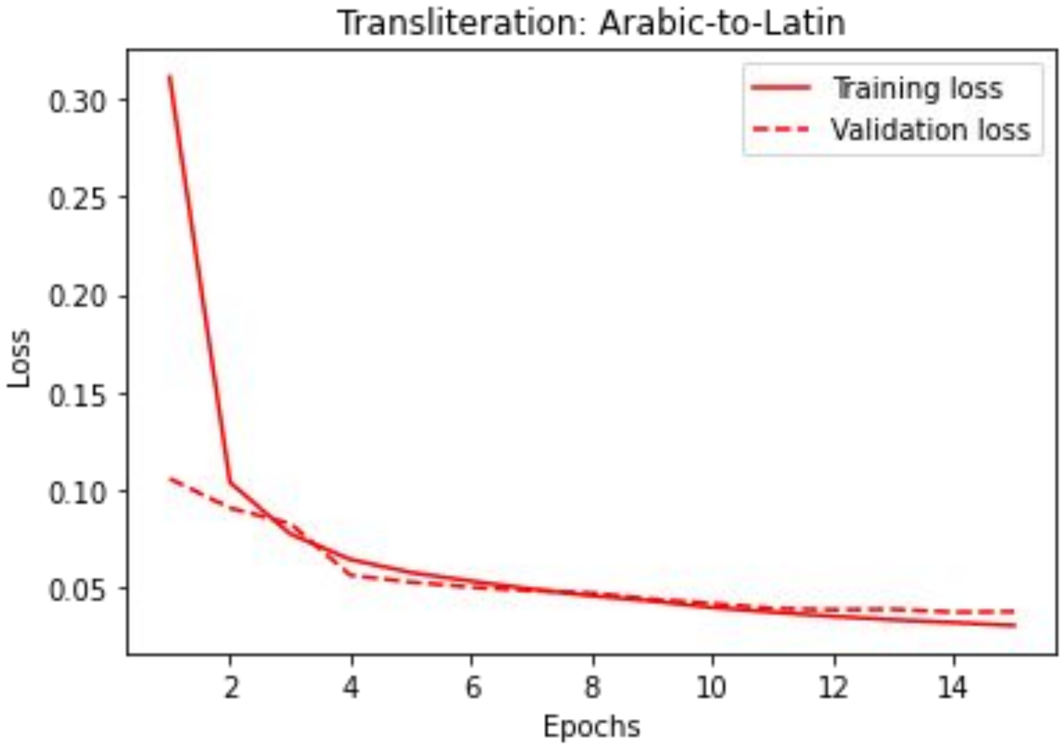
0	1	0	0	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n
0	0	0	1	0	0	...	n

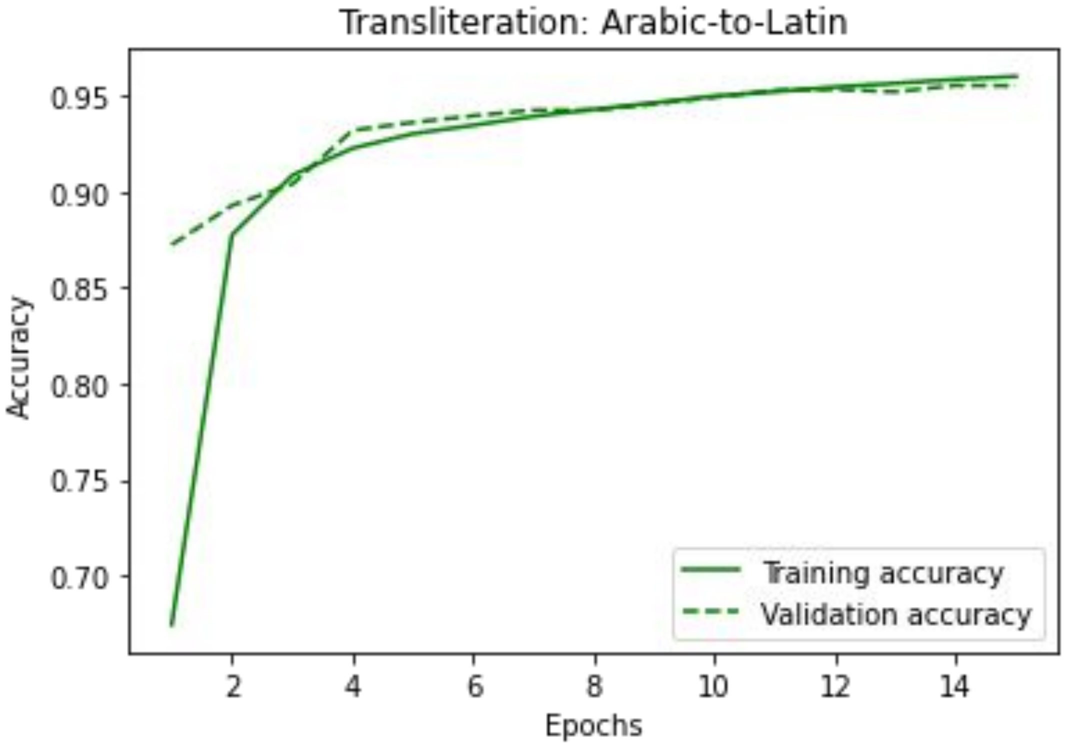
Positional Embedding

1	0	0	0	0	0	...	n
0	1	0	0	0	0	...	n
0	0	1	0	0	0	...	n
0	0	0	1	0	0	...	n

```
{atzmon,
,...
,...
,...}
```

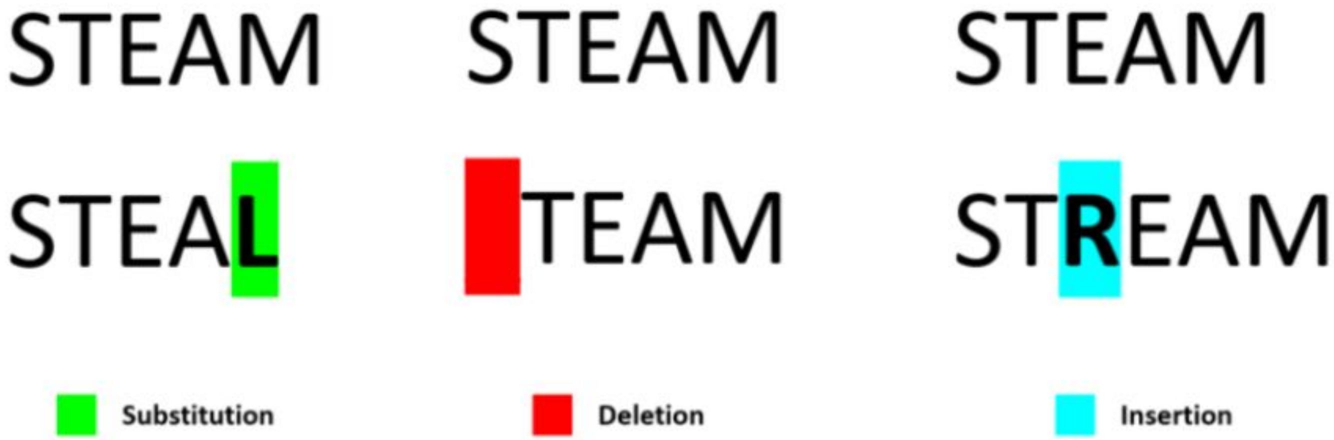
Model training





Measuring accuracy: Edit distance

- **Substitution error**: Misspelled characters/words
- **Deletion error**: Lost or missing characters/words
- **Insertion error**: Incorrect inclusion of character/words



Measuring accuracy: Character error rate

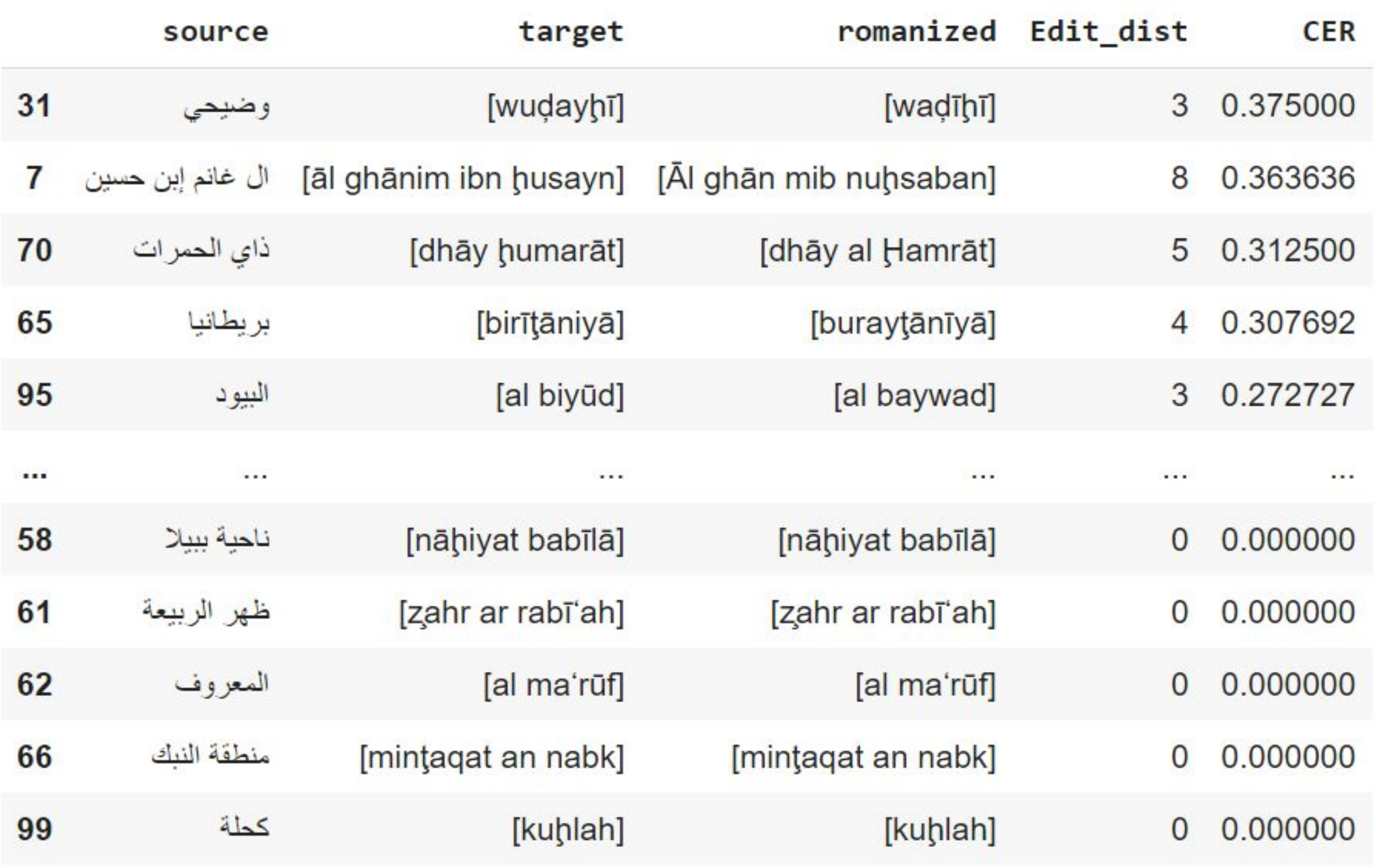
- **S** = Number of **S**ubstitutions
- **D** = Number of **D**eletions
- **I** = Number of **I**nsertions
- **N** = **N**umber of characters in reference text (aka ground truth)

$$CER = \frac{S + D + I}{N}$$

The output of this equation represents the **percentage** of characters in the reference text that was **incorrectly** predicted in the OCR output. The lower the CER value (with **0** being a perfect score), the better the performance of the OCR model.

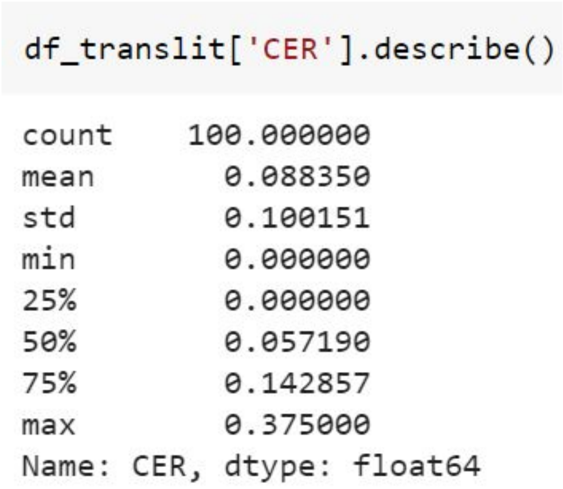
We repeat this calculation for all the pairs of transcribed output and corresponding ground truth, and **take the mean** of these values to obtain an overall CER percentage.

Results



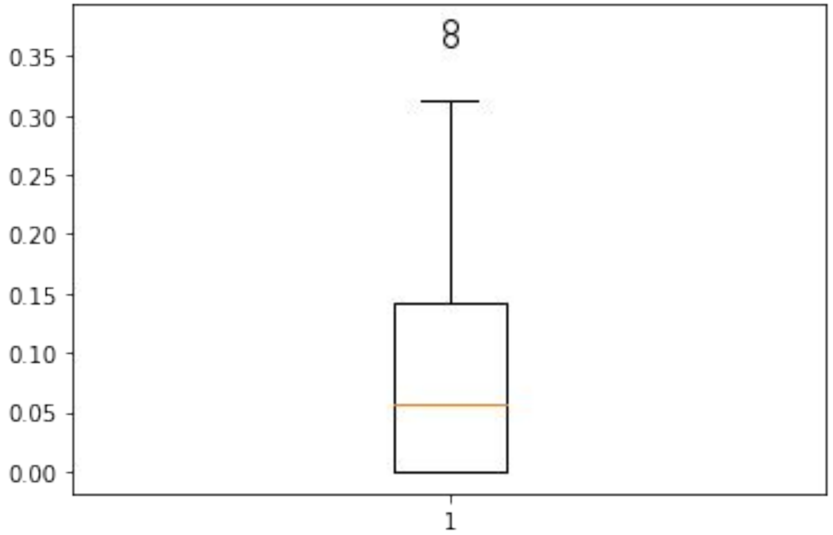
	source	target	romanized	Edit_dist	CER
31	وضيحي	[wuḍayḫī]	[waḍīḫī]	3	0.375000
7	ال غانم إبن حسين	[āl ghānim ibn ḫusayn]	[Āl ghān mib nuḫsaban]	8	0.363636
70	ذاي الحمرات	[dhāy ḫumarāt]	[dhāy al Ḫamrāt]	5	0.312500
65	بريطانيا	[birīţāniyā]	[burayţānīyā]	4	0.307692
95	البيود	[al biyūd]	[al baywad]	3	0.272727
...	...	...	...	...	...
58	ناحية ببيلا	[nāḫiyat babīlā]	[nāḫiyat babīlā]	0	0.000000
61	ظهر الربيعة	[z̧ahr ar rabīʻah]	[z̧ahr ar rabīʻah]	0	0.000000
62	المعروف	[al maʻrūf]	[al maʻrūf]	0	0.000000
66	منطقة النبك	[minţaqat an nabk]	[minţaqat an nabk]	0	0.000000
99	كحلة	[kuḫlah]	[kuḫlah]	0	0.000000

Code Demo: Pandas accuracy analysis

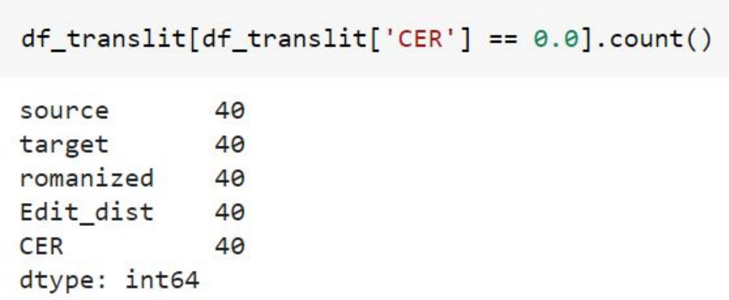


```
df_translit['CER'].describe()
```

```
count    100.000000
mean       0.088350
std        0.100151
min        0.000000
25%        0.000000
50%        0.057190
75%        0.142857
max        0.375000
Name: CER, dtype: float64
```

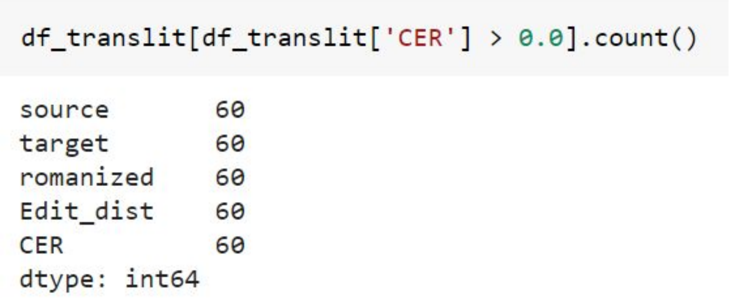


Code Demo: Pandas accuracy analysis



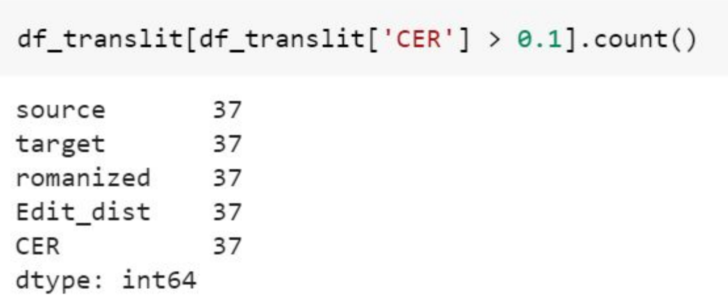
```
df_translit[df_translit['CER'] == 0.0].count()
```

```
source       40
target       40
romanized    40
Edit_dist    40
CER          40
dtype: int64
```



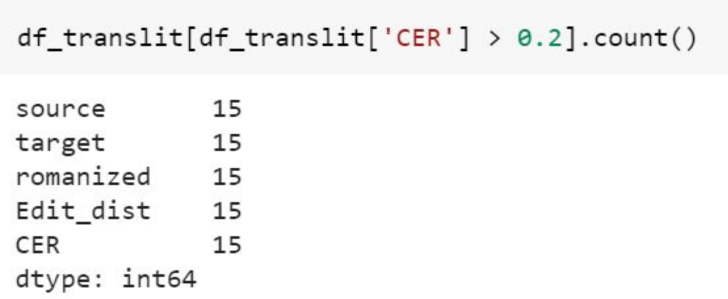
```
df_translit[df_translit['CER'] > 0.0].count()
```

```
source       60
target       60
romanized    60
Edit_dist    60
CER          60
dtype: int64
```



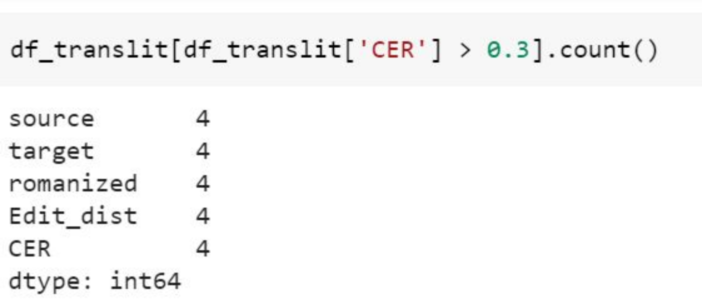
```
df_translit[df_translit['CER'] > 0.1].count()
```

```
source       37
target       37
romanized    37
Edit_dist    37
CER          37
dtype: int64
```



```
df_translit[df_translit['CER'] > 0.2].count()
```

```
source       15
target       15
romanized    15
Edit_dist    15
CER          15
dtype: int64
```



```
df_translit[df_translit['CER'] > 0.3].count()
```

```
source       4
target       4
romanized    4
Edit_dist    4
CER          4
dtype: int64
```

Benefits of transliteration

- (U) Provide transliteration solution for speciality languages
- (U) Automate process of programmatic rule-based solution

NATIONAL GEOSPATIAL-INTELLIGENCE AGENCY
UNITED STATES OF AMERICA