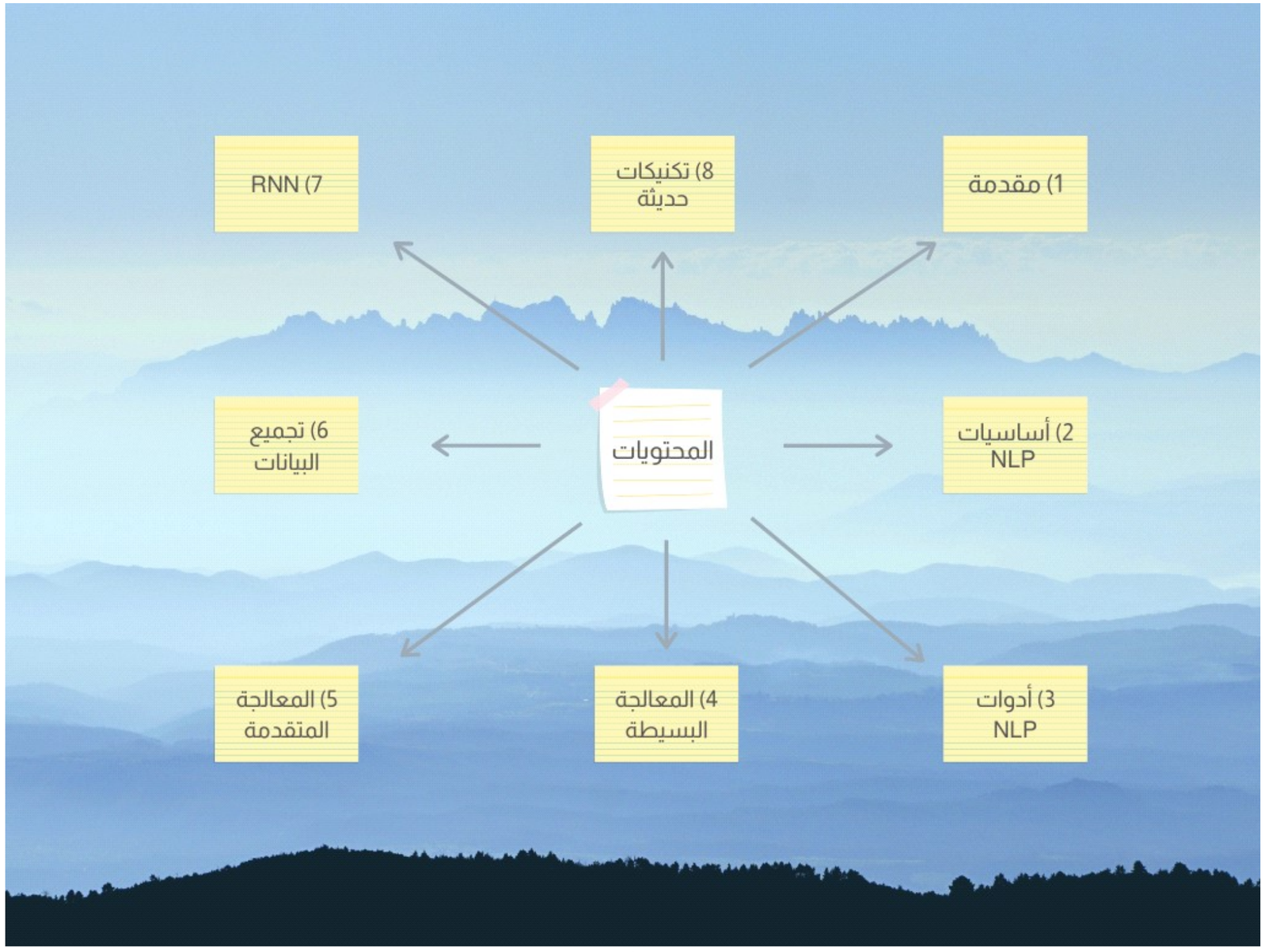
Natural Language Processing المعالجة اللغوية الطبيعية



المحتويات

1) مقدمة	المحتويات	ما هو NLP	تاريخ NLP	العقبات و التحديات	التطبيقات				
2) أساسيات NLP	المكتبات	الملفات النصية	ملفات pdf	البحث في النصوص					
3) أدوات NLP	Tokenization	Sent. Segm.	POS	Stem & Lemm	NER	Stopwords	Matchers	Syntactic Struc.	T.Visualization
4)المعالجة البسيطة	Word Meaning	Word embed	T. Vectors	Word2Vec	BOW	TF-IDF	Text Similarity	Dist. Similarity	
5)المعاجلة المتقدمة	T. Classification	T. Clustering	LDA	NMF	L. Modeling	GloVe	Lexicons	NGrams	T. Generation
	Sent. Analysis	Naïve Bayes	Vader	Auto Correct	Ans. Questions		Summarization & Snippets		
6)تجميع البيانات	Tweet Collecting	Data Scraping	Information Extraction		Information Retrieval		Relative Extraction		Search Engine
RNN (7	Seq to Seq	LSTM	GRU	Rec NN\TNN					
8)تكنيكات حديثة	Word Cloud	CNN	T. Forcing	Attention Model	Transformer	Bert	FastText	Gensim	Chat Bot

القسم الثالث : أدوات NLP

الجزء الأول : Tokenization

==========

الترميز Tokenization
وهي عملية تقسيم الجملة الي عدد من الأجزاء (الكلمات) و تسمي ل واحدة token أو يعني فصل كل كلمة علي حدة , للتعامل معها و معرفة نوعها و ما الي ذلك

و هي قائمة علي فصل الكلمات من الجمل , بحيث تكون كل كلمة وحدها , وهناك نوعين منها : Word tokenizer اي فصل الكلمات , و sentence tokenizer اي فصل الجمل كل جملة علي حدة

وفكرة Word tokenizer قريبة من امر split , لكن مع بعض الاختلاف , فأمر split قائم علي وجود المسافة او الرمز الذي سيتم الفك من خلاله , لكن tokenize قائمة علي معني الكلمة , و غالبا ما يظهر الاختلاف لدي الكلمات الملتصقة ببعضها و لكن بمعني مختلف , مثل I`m او I`d او علامات الاستفهام ,

وهو الخاص بتقسيم الجملة الي كلمات متفرقة , ويجب ان نراعي فيها أكثر من نقطة :

- ان الالجوريثم يكون غالبا ماهرا في فصل الكلمات المختلفة عن بعضها l`d تتحول الي l + ‘d
- كذلك يكون ماهرا في ضم الكلمات المتعلقة ببعضها los angeles هي كلمة واحدة و ليست كلمتين
- تقابله بعض الصعوبات أحيانا , ففي اللغة الفرنسية تتحول L’ الي un

- ***L'ensemble*** → one token or two?
- ***L ? L’ ? Le ?***
- Want ***l’ensemble*** to match with ***un ensemble***

و في الالمانية نجد صعوبة في فك الكلمات الملتصقة معا

German noun compounds are not segmented

- ***Lebensversicherungsgesellschaftsangestellter***
- 'life insurance company employee'

و في اليابانية و الصينية لا توجد مسافات بين الكلمات

- 莎拉波娃现在居住在美国东南部的佛罗里达。
- 莎拉波娃 现在 居住 在 美国 东南部 的 佛罗里达
- Sharapova now lives in US southeastern Florida

لذا تستخدم معهما طريقة max-match و التي تعني : بدأ من بداية الجملة و نحدد اقصي طول يمكن ان يؤدي الي معني معين , فاذا وصلنا اليه , نبدا من الحرف التالي و نكرر العملية

فمثلا الجملة التالية تم تقسيمها هكذا

Thecatinthehat	the cat in the hat

لكن الاسلوب ليس ناجحا دائما في الانجليزية لان هذه الجملة خطأ

Thetabledownthere	the table down there theta bled own there

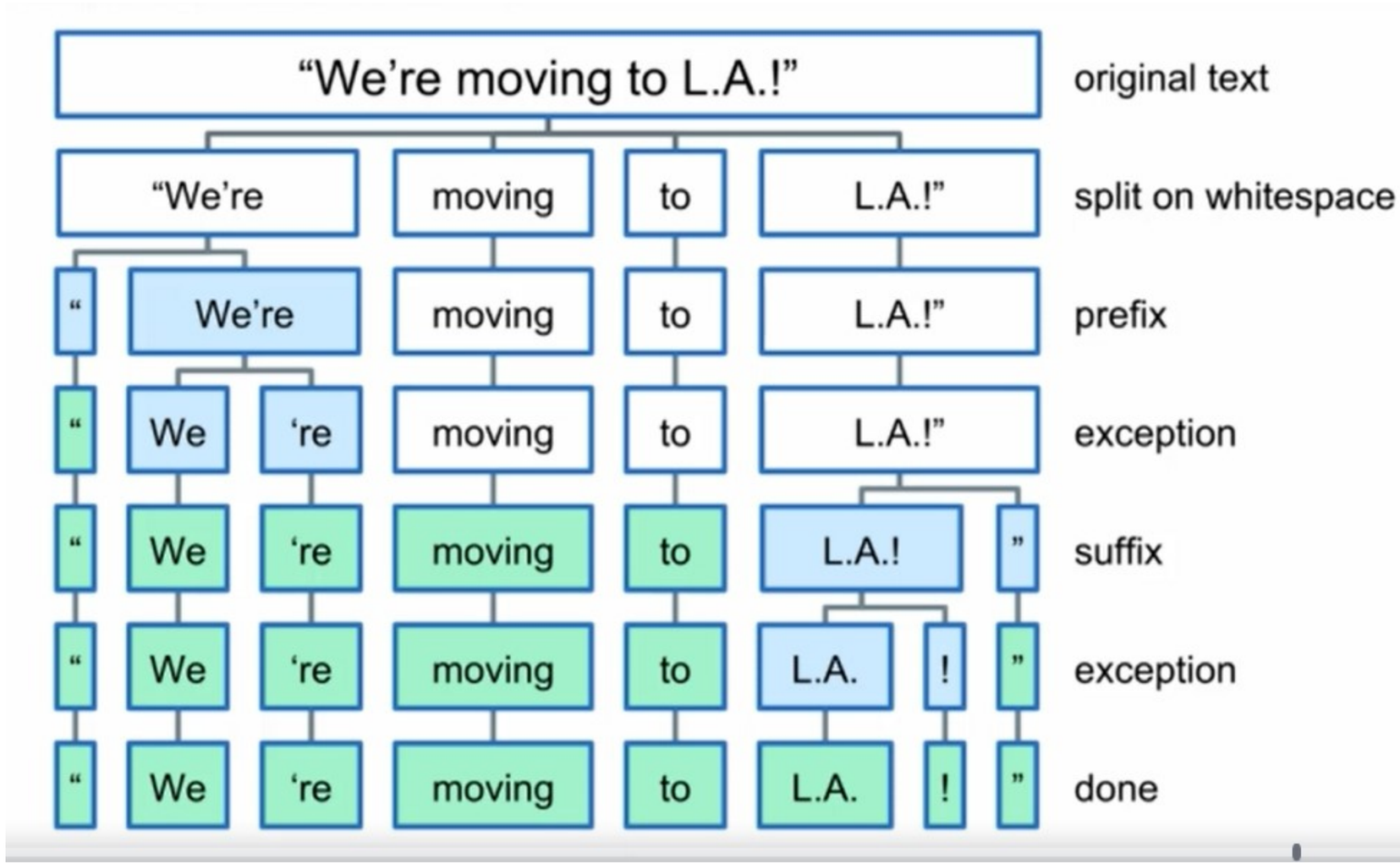
لكن له نتائج ايجابية في الصينية و اليابانية

_

و يتم الأمر علي درجات او مراحل , فلو كان لدينا جملة هي :

“We`re moving to L.A. !”

فيتم تقسيمها حسب الفراغات , ثم حسب الاقواس او الكوتيشين , ثم الابسوتروف , ثم علامات الترقيم و هكذا



و بالطبع هذه الخطوات لا تكون بالترتيب و بشكل الزامي , بل هي حسب كل جملة علي حدة

-

و لبدأ عمل تطبيق عملي بسيط يمكن استخدام الكود المبسط فهنا نقوم باستدعاء المكتبة , ثم تحميل الملف العام الخاص باللغات

```
import spacy
nlp = spacy.load('en_core_web_sm')
```

و هنا نقوم باعطاءه جملة ما ليقوم بالتعرف عليها

```
doc = nlp('Tesla is looking at buying U.S. startup for $6 million')

for token in doc:
    print(token.text)
    print(token.shape_)
```

```
print(token.is_alpha)
print(token.is_stop)
print('---------------')
```

ثم عبر فور , نقوم بجعله يقوم بتحليل كل كلمة علي حدة

هنا يقوم بفصل كل كلمة عن كلمة , ويكون فصل ذكي , فمع ان 6$ ملتصقة الا انه فهم ان كل كلمة لوحدها

و يقوم ايضا بتحديد شكل الكلمة , وهل الكلمة هي حروف ام ارقام , وهل هي من stopwords ام لا

و يمكن استعراض الكلمات هكذا :

```
doc[0] , doc[1] , doc[2] , doc[3] , doc[4] , doc[5] , doc[6] , doc[7]
```

مثال آخر لاستقطاع كلمات معينة من النص

```
doc2 = nlp('''
Although commmonly attributed to John Lennon from his song "Beautiful Boy",
```

```
the phrase "Life is what happens to us while we are making other plans" was written by
cartoonist Allen Saunders and
published in Reader\'s Digest in 1957, when Lennon was 17.
"")

life_quote = doc2[16:30]
print(life_quote)
```

-

مثال آخر :

```
mystring = '"We\'re moving to L.A.!"'
print(mystring)

doc3 = nlp(mystring)
```

```
for token in doc3:
    print(token.text, end=' | ')
```

أمثلة أخري

```
doc4 = nlp(u"We're here to help! Send snail-mail, email support@oursite.com or visit us at http://www.oursite.com!")

for token in doc4:
    print(token)


doc5 = nlp(u'A 5km NYC cab ride costs $10.30')

for token in doc5:
    print(token)
```

```
doc6 = nlp(u"Let's visit St. Louis in the U.S. next year.")

for token in doc6:
    print(token)
```

-

لكن التوكينز هي فقط للقراءة و غير مسموح بها الكتابة او التعديل

```
doc7 = nlp(u'My dinner was horrible.')
doc8 = nlp(u'Your dinner was delicious.')

# Try to change "My dinner was horrible" to "My dinner was delicious"
doc7[3] = doc8[3]
```

-

أيضا تستخدم أدات : word_tokenize لفصل الكلمات

و نلحظ ان فصل الجمل لم يكن بناء علي الحروف الكابيتال او النقاط (.) , لان كلمة Mr.Smith كان فيها كابيتال و نقطة , لكنه لم يعتبرها جملة منفصلة

```
from nltk.tokenize import  word_tokenize
EXAMPLE_TEXT = """
Hello Mr. Smith, how are you doing today? The weather is great,
and Python is awesome. The sky is pinkish-blue. You shouldn't eat cardboard.
"""
print(word_tokenize(EXAMPLE_TEXT))
```

```
EXAMPLE_TEXT = '''
Thomas Gradgrind, sir.  A man of realities.  A man of facts and calculations.  A man who
proceeds upon the principle that two and two are four, and nothing over, and who is not
to be talked into allowing for anything over.  Thomas Gradgrind, sir—peremptorily
Thomas—Thomas Gradgrind.  With a rule and a pair of scales, and the multiplication
table always in his pocket, sir, ready to weigh and measure any parcel of human nature,
```

and tell you exactly what it comes to. It is a mere question of figures, a case of simple arithmetic. You might hope to get some other nonsensical belief into the head of George Gradgrind, or Augustus Gradgrind, or John Gradgrind, or Joseph Gradgrind (all supposititious, non-existent persons), but into the head of Thomas Gradgrind—no, sir!
In such terms Mr. Gradgrind always mentally introduced himself, whether to his private circle of acquaintance, or to the public in general. In such terms, no doubt, substituting the words ‘boys and girls,’ for ‘sir,’ Thomas Gradgrind now presented Thomas Gradgrind to the little pitchers before him, who were to be filled so full of facts.
Indeed, as he eagerly sparkled at them from the cellarage before mentioned, he seemed a kind of cannon loaded to the muzzle with facts, and prepared to blow them clean out of the regions of childhood at one discharge. He seemed a galvanizing apparatus, too, charged with a grim mechanical substitute for the tender young imaginations that were to be stormed away.
‘Girl number twenty,’ said Mr. Gradgrind, squarely pointing with his square forefinger, ‘I don’t know that girl. Who is that girl?’
'''

print(word_tokenize(EXAMPLE_TEXT))

و هنا اظهار للفرق بين token , split

```
for line in EXAMPLE_TEXT.split('\n')[:20] :
    print(line.split()[:10])
    print('------------------------')
    print(word_tokenize(line)[:10])
    print('==============================================')
```

*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*

و هنا أمثلة للغة العربية

```
***
import spacy
nlp = spacy.load('en_core_web_sm')
```

```
doc = nlp(' يعد الذكاء الإصطناعي من العلوم التي يتسارع التطور فيها بشكل لافت منذ عام 2005 و لمدة 15 سنة ')

for token in doc:
    print(token.text)
    print(token.shape_)
    print(token.is_alpha)
    print(token.is_stop)
    print('---------------')

doc[0] , doc[1] , doc[2] , doc[3] , doc[4] , doc[5] , doc[6] , doc[7]
```

```
doc2 = nlp('''
أبو عبد الله محمد بن موسى الخوارزمي عالم رياضيات وفلك
وجغرافيا مسلم. يكنى باسم الخوارزمي وأبي جعفر. قيل أنه ولد حوالي 164هـ 781م (وهو غير مؤكد) وقيل أنه توفي بعد
232 هـ أي (بعد 847م). يعتبر
من أوائل علماء الرياضيات المسلمين حيث ساهمت أعماله بدور كبير في تقدم الرياضيات في عصره. اتصل بالخليفة العباسي
المأمون وعمل في بيت الحكمة في
بغداد وكسب ثقة الخليفة إذ ولاه المأمون بيت الحكمة كما عهد إليه برسم خارطة للأرض عمل فيها أكثر من سبعين جغرافيا.
قبل وفاته في 850 م/232 هـ
```

```
كان الخوارزمي قد ترك العديد من المؤلفات في علوم الرياضيات والفلك والجغرافيا ومن أهمها كتاب المختصر في حساب الجبر والمقابلة الذي يعد أهم كتبه

")


life_quote = doc2[16:30]
print(life_quote)


doc4 = nlp(u"او تصفح موقع الشركة وهو  info@hp.com يمكنك مراسلتنا علي البريد الإلكتروني للشركة هو
www.hp.com ")

for token in doc4:
    print(token)

from nltk.tokenize import sent_tokenize, word_tokenize

EXAMPLE_TEXT = """
أبو عبد الله محمد بن موسى الخوارزمي عالم رياضيات وفلك
```

وجغرافيا مسلم. يكنى باسم الخوارزمي وأبي جعفر. قيل أنه ولد حوالي 164هـ 781م (وهو غير مؤكد) وقيل أنه توفي بعد 232 هـ أي (بعد 847م). يعتبر
من أوائل علماء الرياضيات المسلمين حيث ساهمت أعماله بدور كبير في تقدم الرياضيات في عصره. اتصل بالخليفة العباسي المأمون وعمل في بيت الحكمة في
بغداد وكسب ثقة الخليفة إذ ولاه المأمون بيت الحكمة كما عهد إليه برسم خارطة للأرض عمل فيها أكثر من سبعين جغرافيا.
قبل وفاته في 850 م/232 هـ
كان الخوارزمي قد ترك العديد من المؤلفات في علوم الرياضيات والفلك والجغرافيا ومن أهمها
"""

```
print(word_tokenize(EXAMPLE_TEXT))

for line in EXAMPLE_TEXT.split('\n')[:20] :
    print(line.split()[:10])
    print('----------------------')
    print(word_tokenize(line)[:10])
    print('==============================================================')
```