

# NLTK: Natural Language Toolkit Overview and Application

Jimmy Lai

[jimmy.lai@oi-sys.com](mailto:jimmy.lai@oi-sys.com)

Software Engineer @ 引京聚點

2012/06/09

# Outline

1. An application based on NLP: 聚寶評
2. Introduction to Natural Language Processing
3. Brief History of NLTK
4. NLTK

# 聚寶評 www.ezpao.com

美食搜尋引擎



店家排行榜

地區/地址/景點

店名/料理/食物，例如：拉麵...

找店家

地圖搜尋

選地點：縣市 > 鄉鎮市區

目前累積 99,575 種食物、2,706,720 筆短評，持續增加中...

搜尋各大部落格食記

# 聚寶評 www.ezpao.com



找美食 搜店家

新竹市

火鍋

找店家

地圖搜尋

選地點：縣市 > 鄉鎮市區

縮小搜尋範圍 (重設)：

本次搜尋：在 **新竹市** 找 **火鍋** 的店家

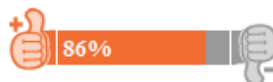
## 語意分析搜尋引擎

約有 93 筆結果

### A 太和殿鴛鴦麻辣火鍋 300 新竹市經國路二段99號

網友分享標籤：綜合鍋類、連鎖店、麻辣火鍋

網友分享菜：鴛鴦麻辣火鍋、綜合菇盤、手工花枝漿 ... 共9道



短評共64筆



個人頭像：網路上也是好評不斷 還有人說比鼎王更好吃 看的我整個熱血沸騰啦! ... 太和殿鴛鴦麻辣火鍋  
2010/07/30

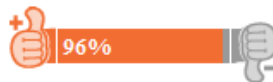
經驗分享：😊 7 😐 0 ☹️ 0

👍 選 0 🗨️ 歌 0

### B 聚北海道昆布鍋 新竹市中正路2號5樓

網友分享標籤：綜合鍋、謝師宴、大型團體聚會、火鍋

網友分享菜：梅子醋、昆布烏龍麵、精緻昆布鍋套餐 ... 共15道



短評共595筆



wensuping：嚴選套餐甜品～金字塔紅豆冰堡，雖然天氣冷，但我還是堅持點這道～好食。 ... 聚-北海道昆布鍋 2011/01/06

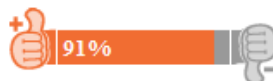
經驗分享：😊 21 😐 0 ☹️ 3

👍 選 0 🗨️ 歌 0

### C 台北林季麻辣火鍋連鎖店 新竹市民生路266號

網友分享標籤：綜合鍋類、麻辣火鍋、吃到飽

網友分享菜：麻辣鍋、鴛鴦鍋、酸白菜鍋、麻辣火鍋 ... 共15道



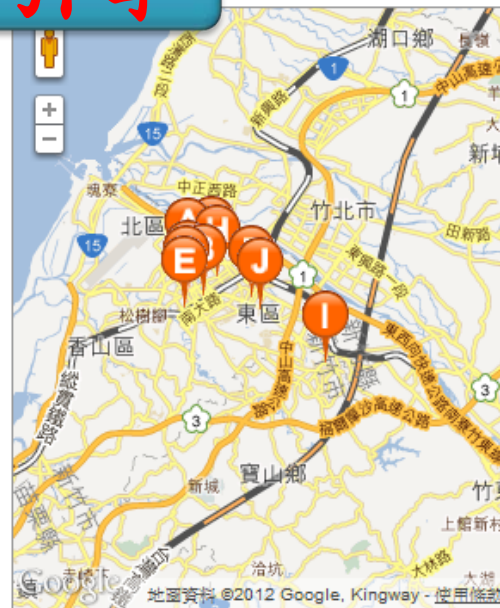
短評共79筆



hisldudi：肉片 / 火鍋類、排骨酥、蛤蜊、鴨血豆腐、香菇丸 這些都挺推薦的。 ... [推薦][新竹市][麻辣火鍋]  
林季麻辣火鍋 (新竹民生店) 2010/03/06

經驗分享：😊 9 😐 0 ☹️ 0

👍 選 0 🗨️ 歌 0



搜尋地點：新竹市

店家排行榜

< 返回 本次搜尋結果

**聚北海道昆布鍋** 新竹市中正路2號5樓 **MAP**

電話：03-5260688

營業時間：

消費價位：一般價位

本店提供的服務：**編輯**

經驗分享： 21 0 3

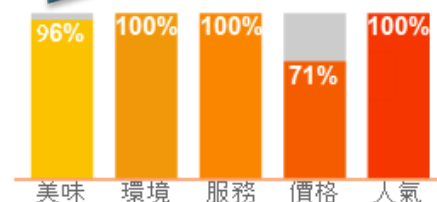
選 0 敬 0

網友分享標籤：綜合鍋、謝師宴、大型團體聚會、節日聚會、綜合鍋類、家庭聚會

網友分享菜：梅子醋(16)、昆布烏龍麵(16)、精緻昆布鍋套餐(13)、北海道金槍奶酪(11)、套餐(11)、蟹肉純龍球(10)、三色魚麵(10)、北海道黃金稀飯(10)、北海道昆布鍋套餐(9)、精緻桂花赤豆饅(9)、金字塔紅豆冰堡(9)、嫩肩牛小排(9)、豬雞雙拼主餐(8)、昆布滋養湯(8)、昆布麻辣湯(8)

## 評論主題分析

註冊來看看朋友對哪些內



短評共595筆

## 網友分享菜分析

短評共 595 筆

搜尋文章內容：



全部主題	全部	內容	全部時間	資料來源
美味		<p>"聚"北海道昆布鍋 - in新竹</p> <p><b>cqvkrpe</b>：↑手工創意單點：蝦仁福袋and山藥玉子燒雙拼 ↑服務人員建議點雙拼的可以吃到兩種不同的東西 山藥玉子燒口感很Q...</p>	2009/12/17 很久以前	隨意窩
魚頭海鮮主餐		<p>新竹-聚北海道昆布鍋</p> <p><b>阿叮</b>：隔壁桌的秀華點了季節魚頭海鮮主餐 端來是大大的一隻魚頭和一些海鮮 看起來不錯吃旁邊還有蝦罵和花枝...</p>	2010/12/07 一年以前	蕃薯藤
美味		<p>[家庭聚餐]生日聚-北海道昆布鍋</p> <p><b>alice</b>：嫩肩牛小排是〔北海道嚴選套餐〕才有的主餐，它真的也沒讓我失望。均勻的油花分在的肉片上，放到鍋中涮幾下就熟了，入口軟嫩，滋味好呀！爸媽主餐選雞肉，雞肉也很不錯；而豬肉是雪花豬，有特別的口感，也很不賴唷～...</p>	2010/12/05 一年以前	新浪部落
北海道昆布鍋		<p>[食·新竹]"聚"在一起吃火鍋</p> <p><b>aorange</b>：北海道昆布鍋清清爽爽的，昆布滋養鍋有中藥的味道，...</p>	2009/10/28 很久以前	愛評網

## 正評/負評分析

# 手機版

m.ezpao.com

## 網友分享菜+評論分析

 [首頁](#)

網友分享菜：辣炒年糕

 看正評  看負評

 小老鼠：拔絲地瓜、海鮮餅、**辣炒年糕**、炸雞、炸柳葉魚都不錯 ...  
[看原文\(壹評網\)](#) - 2009/06/15

 kuopatty：**辣炒年糕**（**辣炒年糕**）年糕對我來說都是大同小異，差別只在於醬式辣醬，這家力辣醬不錯！！） ...  
[看原文\(壹評網\)](#) - 2009/07/01

 littlebear：熟食有絲拔地瓜、海鮮煎餅、炸雞腿、**炒年糕**等~真好吃 ...  
[看原文\(壹評網\)](#) - 2010/05/25

 aliceC：不管是**辣炒年糕**酥炸小雞腿 炸花枝圈 拔絲地瓜 每道菜店家都會不斷補上 保證客人絕對吃到撐破肚皮 **辣炒年糕**炒的超級夠味 完全不會有沒入味的情况 但年糕Q彈口味依舊不會軟爛 ...  
[看原文\(壹評網\)](#) - 2010/05/26

 玉仔：**辣炒年糕**的醬料也很厚實，麻糬也打得很Q很Q。 ...  
[看原文\(壹評網\)](#) - 2010/06/05

 冰淇淋妹：Spring先生說**辣年糕**還不錯 不過 很辣很辣XDD ...  
[看原文\(壹評網\)](#) - 2010/12/09

 vanvan醬：**辣炒年糕**，冷掉吃更有嚼勁 裹上薄薄一層麥芽糖的地瓜鬆鬆綿綿， ...  
[看原文\(壹評網\)](#) - 2009/06/11

 irma5223：小菜第一道--**辣炒年糕**...口感上面還不錯吃 ...  
[看原文\(壹評網\)](#) - 2008/12/10

看更多評論

 [首頁](#)

網友分享菜：辣炒年糕

 看正評  看負評

 蜜峰熊：「**辣炒年糕**」，我覺得還ok感覺有點太硬， ...  
[看原文\(壹評網\)](#) - 2009/06/24

 egeria：**辣年糕**我覺得辣度很够，但年糕煮的有點過軟，吃起來沒有QQ的口感~/ ...  
[看原文\(壹評網\)](#) - 2010/05/25

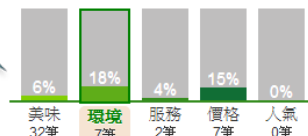
## 網友分享菜+正負評分析

網友分享菜單


海鮮煎餅(49)、拔絲地瓜(47)、辣炒年糕(18)、銅盤烤肉鐵盤烤肉(9)、椒麻雞(10)、銅盤烤肉(10)





社群評論





 看正評  看負評


 詩婷：很推薦吃它的「牛肉」很嫩 很讚 還有它的位子真的很少很少，所以想去吃的朋友們記得要事先打電話預約的唷!!!否則可能會白跑一趟 ...  
[看原文\(壹評網\)](#) - 2010/06/23


 小小庭：覺得這邊的店家應該要比較有質感一點!?韓式算是很樸實的一間店 不過這地板好像不是很乾淨... 然後我想東區地價真的太貴了能運用的空間都要發揮到極致劇所位於廚房的門口 ...  
[看原文\(壹評網\)](#) - 2010/06/25

 sofia1002：櫃台設在這裡，最裡面是廚房，再來是廁所，就是櫃台，其他都是座位區，這裡真的很小一間 ...  
[看原文\(壹評網\)](#) - 2010/05/27

 Alice：愛吃的蛋蛋：這裡的種類不多，但是也夠大家吃了，重點是調味好還蠻好吃的，只是因為空間不太，所以只要有客人大聲那大家的音量就要一起放大，像今天去吃的時候就碰到一組當這家店是自己家的人客 ...  
[看原文\(壹評網\)](#) - 2010/12/22

 優娜的地盤：店小座位少，務必訂位 ...  
[看原文\(痞客邦\)](#) - 2009/11/05

 ~ming~：外面生意真的不錯!話說雖然對店家一直催促+限制2小時的規定讓人覺得吃得很不盡興 還有店內空調實在有待加強 (吃完回家 全身上下都是嚴重油煙味然後不知為何燒烤過程湯汁噴得很厲害) ...  
[看原文\(壹評網\)](#) - 2009/07/18

 joanne.chen：食材飲料區(其實我覺得東西並不多...可能是店內空間也真的不大)不定時地推出熟食 ...  
[看原文\(隨意窩\)](#) - 2009/08/26

# Natural Language Processing (NLP)

- 語音識別 (Speech recognition)
- 詞性標註 (Part-of-speech tagging)
- 句法分析 (Parsing)
- 自然語言生成 (Natural language generation)
- 文本分類 (Text classification)
- 信息抽取 (Information extraction)
- 機器翻譯 (Machine translation)
- 文字蘊涵 (Textual entailment)

via Wikipedia

# NLTK: Natural Language Toolkit

- <http://www.nltk.org/>
- Author: Steven Bird, Edward Loper, Ewan Klein
- Originally developed for class student has background either in computer science or linguistics.
- Currently:
  - Education: over 100 courses in 23 countries.
  - Research: over 250 papers cites NLTK.



# Outline

1. An application based on NLP: 聚寶評
2. Introduction to Natural Language Processing
3. Brief History of NLTK
4. NLTK:
  - a. Installation
  - b. Annotated Text Corpora
  - c. Text Tokenization, Normalization, Analysis, Distribution Analysis
  - d. Part-of-speech Tagging
  - e. Text Classification
  - f. Named Entity Recognition

# Install NLTK

Python 2.6+

`pip install numpy`

`pip install matplotlib`

`pip install nltk`

# Annotated Text Corpora (1/3)

## nlTK.corpus

- Corpus: 語料庫，含有某種結構化標記的資料集合，可能包含多種語言。
- 例如：
  - stopwords: 常見字字典
  - sinica\_treebank: 中文語句結構標記語料庫
  - brown: 包含15種分類及詞性標記的英語語料庫
  - wordnet: 包含詞性、同義反義的英語字典

# Annotated Text Corpora (2/3)

## nlTK.corpus

```
import nlTK
```

```
#download corpus on demand
```

```
nlTK.download()
```

```
#stopwords
```

```
nlTK.corpus.stopwords.words()
```

```
nlTK.corpus.stopwords.words('english')
```

```
nlTK.corpus.stopwords.words('french')
```

```
some_english_stopwords = ['most', 'me',  
'below', 'when', 'which', 'what', 'of', 'it',  
'very', 'our']
```

```
#Chinese treebank
```

```
nlTK.corpus.sinica_treebank
```

```
#Examples
```

```
(嘉珍, Nba)
```

```
(和, Caa)
```

```
(我, Nhaa)
```

```
(住在, VC1)
```

```
(同一條, DM)
```

```
(巷子, Nab)
```

```
(我們, Nhaa)
```

```
(是, V_11)
```

```
(鄰居, Nab)
```

```
(也, Dbb)
```

```
(是, V_11)
```

```
(同班, Nv3)
```

```
(同學, Nab)
```

# Annotated Text Corpora (3/3)

## nlTK.corpus

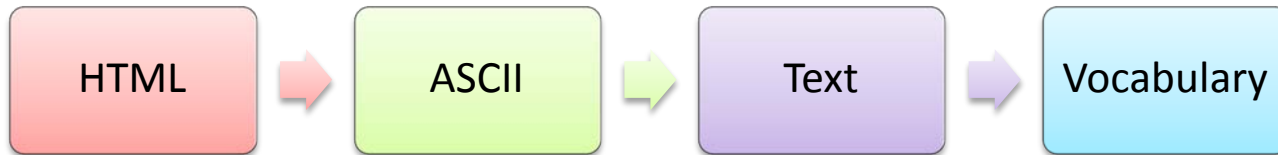
ACE Named Entity Chunker (Maximum entropy)  
Australian Broadcasting Commission 2006  
Alpino Dutch Treebank  
BioCreAtIvE (Critical Assessment of Information  
Extraction Systems in Biology)  
Brown Corpus  
Brown Corpus (TEI XML Version)  
CESS-CAT Treebank  
CESS-ESP Treebank  
Chat-80 Data Files  
City Database  
The Carnegie Mellon Pronouncing Dictionary (0.6)  
ComTrans Corpus Sample  
CONLL 2000 Chunking Corpus  
CONLL 2002 Named Entity Recognition Corpus  
Dependency Treebanks from CoNLL 2007 (Catalan  
and Basque Subset)  
Dependency Parsed Treebank  
Sample European Parliament Proceedings Parallel  
Corpus

Portuguese Treebank  
Gazeteer Lists  
Genesis Corpus  
Project Gutenberg Selections  
NIST IE-ER DATA SAMPLE  
C-Span Inaugural Address Corpus  
Indian Language POS-Tagged Corpus  
JEITA Public Morphologically Tagged Corpus (in  
ChaSen format)  
PC-KIMMO Data Files  
KNB Corpus (Annotated blog corpus)  
Language Id Corpus  
Lin's Dependency Thesaurus  
MAC-MORPHO: Brazilian Portuguese news text with  
part-of-speech tags  
Machado de Assis -- Obra Completa  
Sentiment Polarity Dataset Version 2.0  
Names Corpus, Version 1.3 (1994-03-29)  
NomBank Corpus 1.0  
NPS Chat  
Paradigm Corpus

# Text Tokenization

## nlk.tokenize

### Web Text Processing Flow



```
from urllib import urlopen
html = urlopen(url).read()
raw = nltk.clean_html(html)
sents = nltk.sent_tokenize(raw)
tokens = []                                # wordpunct_tokenize: ['3', '.', '33']
for sent in sents:                         # word_tokenize: ['3.33']
    tokens.extend(nltk.word_tokenize(sent))
text = nltk.Text(tokens)
words = [word.lower() for word in text]
vocab = sorted(set(words))
```

# Text Normalization (1/2)

## nlTK.stem

- Stem: 將單字(現在式、過去式、單複數)還原成原型。可以將不同形式的單字歸類為同一個單字。
- 著名演算法：Porter Stemmer

# Text Normalization (2/2)

## nltk.stem

```
In [1]: words = nltk.corpus.conll2000.words()[1:20]
```

```
In [2]: print words
```

```
['Confidence', 'in', 'the', 'pound', 'is', 'widely', 'expected', 'to', 'take', 'another',  
'sharp', 'dive', 'if', 'trade', 'figures', 'for', 'September', ',', 'due', 'for']
```

```
In [3]: stemmer = nltk.stem.PorterStemmer()
```

```
In [4]: print [stemmer.stem(word) for word in words]
```

```
['Confid', 'in', 'the', 'pound', 'is', 'wide', 'expect', 'to', 'take', 'anoth',  
'sharp', 'dive', 'if', 'trade', 'figur', 'for', 'Septemb', ',', 'due', 'for']
```

```
In [5]: stemmer = nltk.stem.LancasterStemmer()
```

```
In [6]: print [stemmer.stem(word) for word in words]
```

```
['confid', 'in', 'the', 'pound', 'is', 'wid', 'expect', 'to', 'tak', 'anoth',  
'sharp', 'div', 'if', 'trad', 'fig', 'for', 'septemb', ',', 'due', 'for']
```

```
In [7]: stemmer = nltk.stem.WordNetLemmatizer()
```

```
In [8]: print [stemmer.lemmatize(word) for word in words]
```

```
['Confidence', 'in', 'the', 'pound', 'is', 'widely', 'expected', 'to', 'take', 'another',  
'sharp', 'dive', 'if', 'trade', 'figure', 'for', 'September', ',', 'due', 'for']
```



# Text Analysis

## nlk.text

```
#using sinica treebank dataset
In [1]: text = nltk.Text(nltk.corpus.sinica_treebank.words())

#common context
In [2]: text.common_contexts(['是', '在'])
一直_我_也_一_也_台北_也_她_他_一個_就_大_就_我們
就_這_的_他_說_中國_都_樹

#collocation
In [3]: text.collocations()
Building collocations list
全 國; 國 家 公 園; 就 是; 身 上; 都 是; 才 能; 站 在;
還 有; 最 大; 河 邊; 的 時 候; 變 得; 心 裡; 這 是; 各
地; 帶 著; 院 子 裡; 一 個 人; 是 一 個; 就 會

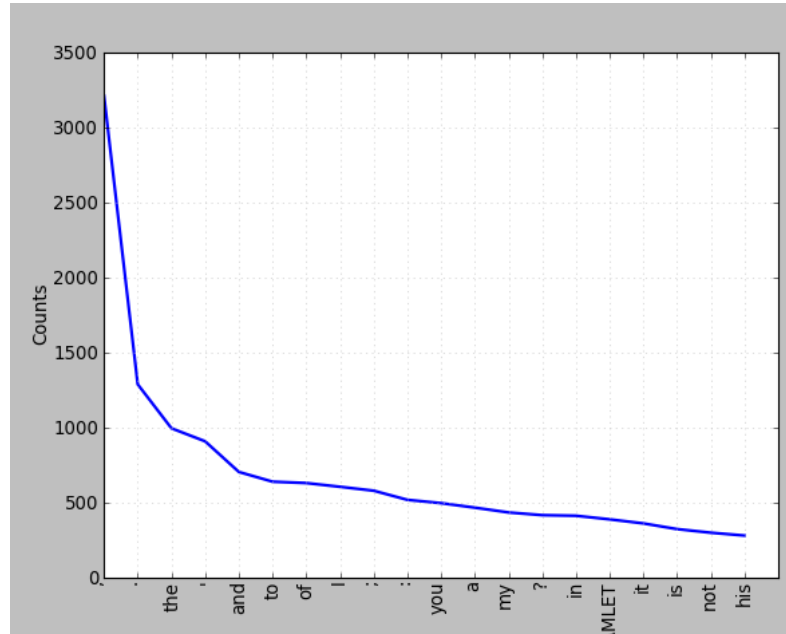
#concordance
In [4]: text.concordance('台灣', lines=10)
Displaying 10 of 161 matches:
?? I B M C l u b 的 組 織 台 灣 I B M 為 配 合 國 內 勞 工
?? 男 子 獨 專 成 功 立 名 的 台 灣 為 女 性 揚 眉 吐 氣 她 們 是
?? 智 慧 的 她 女 性 形 象 在 台 灣 和 中 國 大 陸 小 說 是 解 放
腐 的 同 時 在 雙 十 節 宣 稱 台 灣 要 以 統 一 大 業 為 己 任 ?
資 源 的 條 件 因 為 今 天 的 台 灣 並 沒 有 發 展 出 一 個 具 備
有 一 位 廣 播 電 視 廳 長 是 台 灣 基 隆 人 一 個 人 如 果 已 ?
?? 粗 俗 的 水 泥 步 道 時 在 台 灣 經 濟 發 展 的 歷 史 赤 裸 裸
?? 的 自 然 的 力 量 為 避 免 台 灣 原 住 民 族 文 化 特 質 與 總 ?
?? 註 明 清 時 期 漢 人 嫁 給 台 灣 平 埔 族 女 子 大 部 份 摻 雜 ?
國 報 紙 上 看 到 一 則 介 紹 台 灣 近 況 的 報 導 尋 找 自 我 認
```

# Text Distribution Analysis

## nltk.probability

```
In [1]: import nltk
In [2]: dist = nltk.probability.FreqDist(nltk.corpus.shakespeare.words('hamlet.xml'))
In [3]: dist.tabulate(20)
, . the ' and to of I ; : you a my ? in HAMLET it is not his
3211 1289 996 909 705 640 631 606 580 519 497 467 435 417 413 389 362 324 300 281
In [4]: dist.plot(20)
In [5]: dist['the']
Out[5]: 996

In [6]: cdist = nltk.probability.ConditionalFreqDist(
        |         (len(word), word) for word in nltk.corpus.shakespeare.words('hamlet.xml'))
In [7]: float(cdist[3]['the']) / float(cdist[3].N())
Out[7]: 0.14619110523998238
```



# Part-of-speech Tagging (1/3)

## nlk.tag

- Part of Speech Tagging: 詞性標記，標記每個單字的詞性。
- 同一單字的不同詞性其語義不同，如Book名詞是書，動詞是預定。
- 透過POS Tagging，可以賦予文字更多語義資訊。

Tag	Meaning
ADJ	adjective
ADV	adverb
CNJ	conjunction
DET	determiner
EX	existential
FW	foreign word
MOD	modal verb
N	noun
NP	proper noun
NUM	number
PRO	pronoun
P	preposition

# Part-of-speech Tagging (2/3)

## nlk.tag

```
In [1]: print nltk.corpus.brown.tagged_words()[:15]
[('The', 'AT'), ('Fulton', 'NP-TL'), ('County', 'NN-TL'), ('Grand', 'JJ-TL'),
('Jury', 'NN-TL'), ('said', 'VBD'), ('Friday', 'NR'), ('an', 'AT'),
('investigation', 'NN'), ('of', 'IN'), ('Atlanta's', 'NP$'), ('recent', 'JJ'),
('primary', 'NN'), ('election', 'NN'), ('produced', 'VBD')]
```

#an off-the-shelf tagger

```
In [2]: print nltk.tag.pos_tag(nltk.corpus.brown.words()[:15])
[('The', 'DT'), ('Fulton', 'NNP'), ('County', 'NNP'), ('Grand', 'NNP'), ('Jury', 'NNP'),
('said', 'VBD'), ('Friday', 'NNP'), ('an', 'DT'), ('investigation', 'NN'), ('of', 'IN'),
('Atlanta's', 'JJ'), ('recent', 'JJ'), ('primary', 'JJ'), ('election', 'NN'),
('produced', 'VBN')]
```

# Part-of-speech Tagging (3/3)

## nlk.tag

```
#train our tagger
In [3]: unigram_tagger = nltk.tag.UnigramTagger(
        |         |         nltk.corpus.brown.tagged_sents(categories='news')[:1000])
In [4]: unigram_tagger.evaluate(nltk.corpus.brown.tagged_sents(categories='news')[1000:2000])
Out[4]: 0.6709000287549123

In [5]: affix_tagger = nltk.tag.AffixTagger(
        |         |         nltk.corpus.brown.tagged_sents(categories='news')[:1000], backoff=unigram_tagger)
In [6]: affix_tagger.evaluate(nltk.corpus.brown.tagged_sents(categories='news')[1000:2000])
Out[6]: 0.6848461612192083

In [7]: bigram_tagger = nltk.tag.BigramTagger(
        |         |         nltk.corpus.brown.tagged_sents(categories='news')[:1000], backoff=affix_tagger)
In [8]: bigram_tagger.evaluate(nltk.corpus.brown.tagged_sents(categories='news')[1000:2000])
Out[8]: 0.7011406115211348
```

# Text Classification (1/3)

## `nltk.classify`

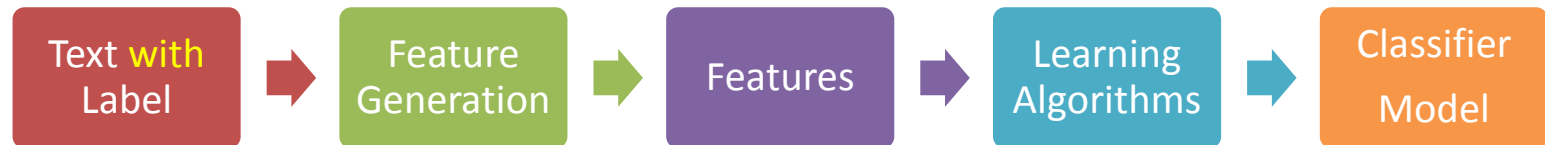
- Text Classification: 文字分類，分析文字後將文字分到預先定義的類別裡。
- 基於統計的機器學習演算法，著名的演算法為：
  - Naïve Bayes Classifier
  - Decision Tree
  - Support Vector Machine

# Text Classification (2/3)

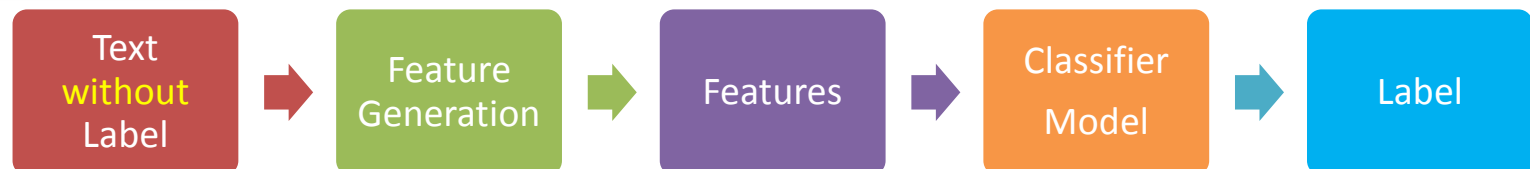
`nltk.classify`

## Machine Learning Approach Work Flow

### Training



### Prediction



# Text Classification (3/3)

## nltk.classify

```
In [1]: name_label_list = []
In [2]: name_label_list.extend([ (name, 'female') for name in nltk.corpus.names.words('female.txt')])
In [3]: name_label_list.extend([ (name, 'male') for name in nltk.corpus.names.words('male.txt')])
In [4]: random.shuffle(name_label_list)
In [5]: name_label_list[:5]
Out[6]:
[('Giffard', 'male'),
 ('Melosa', 'female'),
 ('Minerva', 'female'),
 ('Debora', 'female'),
 ('Adrien', 'male')]

In [7]: def get_feature(name):
...:     features = {'first_letter':name[0], 'last_letter':name[-1],
...:                 'first_2_letter': name[:2], 'last_2_letter': name[-2:],
...:                 'first_3_letter': name[:3], 'last_3_letter': name[-3:]}
...:     return features
...:

In [8]: feature_set = [(get_feature(name), label) for (name, label) in name_label_list ]
In [9]: size = int(len(feature_set) * 0.9)
In [10]: train_set, test_set = feature_set[:size], feature_set[size:]
In [11]: classifier = nltk.classify.NaiveBayesClassifier.train(train_set)
In [12]: nltk.classify.accuracy(classifier, test_set)
Out[13]: 0.8377358490566038
```

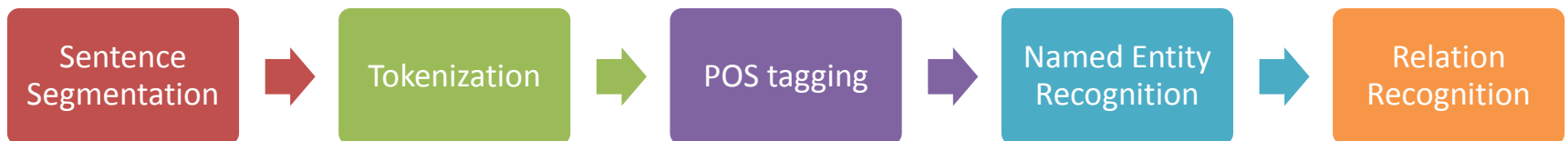


# Named Entity Recognition (NER) (1/2)

`nltk.tag`, `nltk.chunk`

- Named Entity Recognition: 從文字中擷取出命名實體，命名實體是具有完整語義的複合單字。例如：人名、地名、事件。

## NER General Work Flow



# Named Entity Recognition (NER) (2/2)

## nlk.tag, nltk.chunk

```
In [1]: import nltk
In [2]: tagger = nltk.tag.stanford.NERTagger(
        |         |         | 'stanford-ner-2012-05-22/classifiers/english.conll.4class.distsim.crf.ser.gz',
        |         |         | 'stanford-ner-2012-05-22/stanford-ner-2012-05-22.jar')
In [3]: tagger.tag('Rami Eid is studying at Stony Brook University in NY'.split())
Out[3]:
[('Rami', 'PERSON'),
 ('Eid', 'PERSON'),
 ('is', 'O'),
 ('studying', 'O'),
 ('at', 'O'),
 ('Stony', 'ORGANIZATION'),
 ('Brook', 'ORGANIZATION'),
 ('University', 'ORGANIZATION'),
 ('in', 'O'),
 ('NY', 'O')]
```

# Reference

1. Steven Bird, Ewan Klein, and Edward Loper, “**Natural Language Processing with Python**”, 2009. #includes: Python + NLP + NLTK
2. Jacob Perkins, “**Python Text Processing with NLTK 2.0 Cookbook**”, 2010.
3. Matthew A. Russell, “Mining the Social Web”, 2011.
4. Loper, E., & Bird, S. (2002). NLTK: The Natural Language Toolkit. Proceedings of the ACL02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics.

# Thank you for your attention.

## Q & A

### We are hiring!

- 核心引擎演算法研發工程師
  - 系統研發工程師
  - 網路應用研發工程師
- 本公司提供企業客戶語義分析服務，  
獲創投“軟體銀行”投資與肯定。  
歡迎對創業有熱情的你加入我們！

**Oxygen Intelligence** Taiwan Limited

**引京聚點** 知識結構搜索股份有限公司

- 公司簡介：<http://www.ezpao.com/about/>
- 職缺簡介：<http://www.ezpao.com/join/>
- 請將履歷寄到 jimmy.lai@oi-sys.com