

# บทที่ 4 - NLP คืออะไร บทเรียนจากอดีตสู่ปัจจุบัน

Peerat Limkonchotiwat

PhD student at VISTEC, Thailand

# สารบัญ

เราจะเรียนอะไรกันในบทนี้

NLP คืออะไร?

---

ประวัติศาสตร์ NLP

---

NLP ภาษาไทย

---



# NLP คืออะไร

# NLP คืออะไร

- NLP หรือ Natural Language Processing เป็นสาขาย่อยของภาษาศาสตร์ (Linguistics) และปัญญาประดิษฐ์ (Artificial Intelligence)
- ศึกษาด้านคุณภาพเกี่ยวกับการทำให้คอมพิวเตอร์ “เข้าใจและจัดการ” ภาษาธรรมชาติของมนุษย์ได้
- ยกตัวอย่าง
  - จำแนกประเภทประโยค (Sequence Classification)
    - หนังเรื่องนี้สนุกจังเลย → ประโยคเป็น “บวก”
    - หนังเรื่องนี้แย่มาก → ประโยคเป็น “ลบ”
  - สร้างข้อความ-เติมคำในช่องว่าง (Text Generation)
    - กะเพรา\_\_อร่อยมาก → “กะเพรา หมูสับ อร่อยมาก”, “กะเพรา ไก่ อร่อยมาก”

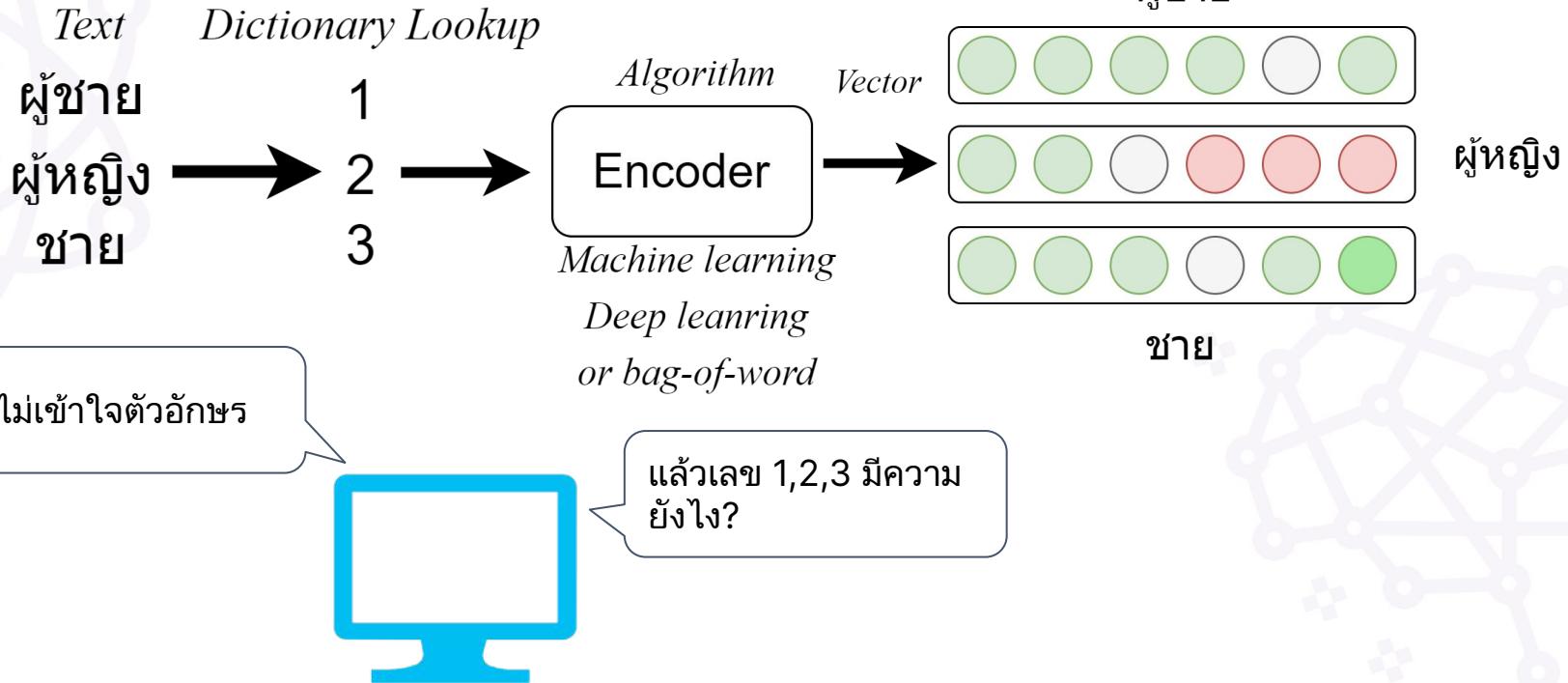
# Challenge?

- คอมพิวเตอร์ไม่ได้เข้าใจ “ภาษาแบบที่มนุษย์เข้าใจ”
  - เช่น สองข้อความนี้แตกต่างกันหรือไม่?
    - “ฉันหิวข้าวมากๆ” และ “ฉันอยากรับประทานอาหารมากๆ”
  - มนุษย์: “ไม่”
  - คอมพิวเตอร์: “ใช่”
    - สาเหตุ: “หิวข้าว” != รับประทานอาหาร



# คอมพิวเตอร์เข้าใจภาษาคน?

# ทำไมคอมพิวเตอร์เข้าใจภาษาคน?

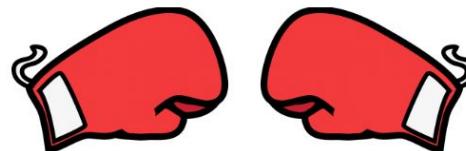




# ประวัติศาสตร์ NLP (แบบง่ายๆสันๆ ไร้คณิตศาสตร์!!)

## ตัดคำ ตัดพยางค์ และทำ Bag-of-Word

- การตัดคำ เป็น task แรกสุดในโปรเซสของการทำ NLP
- หลายภาษาใช้ whitespace ในการแบ่งคำ บางภาษาต้องทำโน้มเดลขึ้นมาเพื่อตัดคำ เช่น ภาษาไทย



# ตาก|ลม VS ตา|กลม

ฉันนั่งตาก|ลมอยู่ริมทะเล

บึงอายทำให้เธอตา|กลมมาก

# ตัดคำ ตัดพยางค์ และทำ Bag-of-Word

หน่วยคำ  
(Word)

วันนี้ฉัน|สั่ง|กิวดัง|มา|ทาน|ที่|บ้าน

หน่วยคำย่อย  
(Subword)

วันนี้|ฉัน|สั่ง|กิ|ว|ด|ัง|มา|ทาน|ที่|บ้าน  
(SentencePiece; XLMR)

หน่วยพยางค์  
(Syllable)

วัน|นี|ฉัน|สั่ง|กิ|ว|ดัง|มา|ทาน|ที่|บ้าน

หน่วยตัวอักษร  
(Character)

ว|า|ร|บ|น|น|เร|ิ|ร|อ|ร|บ|น|ส|า|ร|บ|ง|ก|ิ|ว|ด|ร|บ|ง|มา|ท|า|น|ท|ี|เร|ิ|ร|บ|ร|ก|า|น

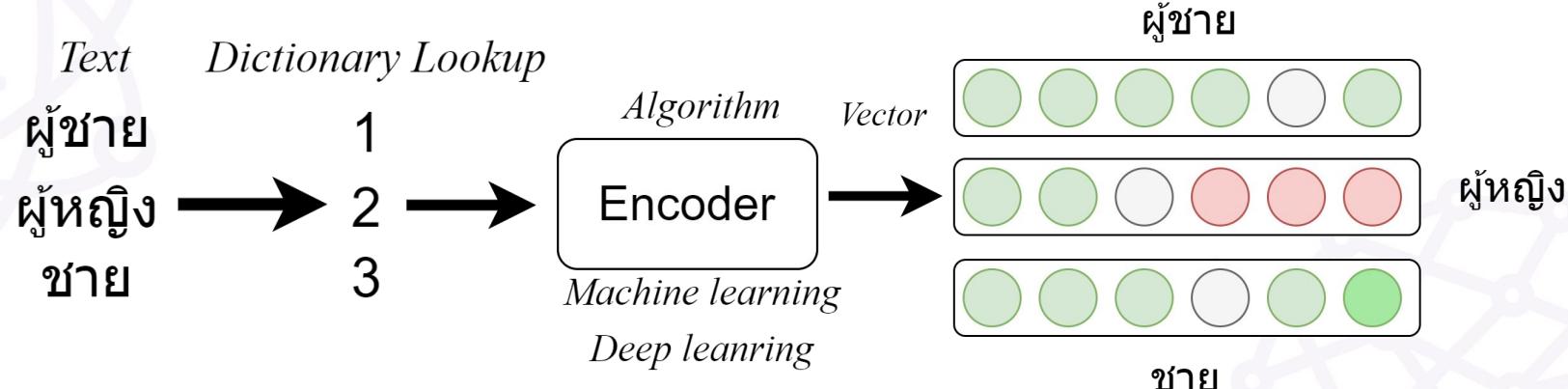
## ตัดคำ ตัดพยางค์ และทำ Bag-of-Word

ประโยค	แมว	นั่ง	มอง	ปลา	บัน	โต๊ะ	กิน
แมว นั่ง มอง ปลา	1	1	1	1	0	0	0
แมว นั่ง บัน โต๊ะ แมว	2	1	0	0	1	1	0
แมว กิน ปลา	1	0	0	1	0	0	1

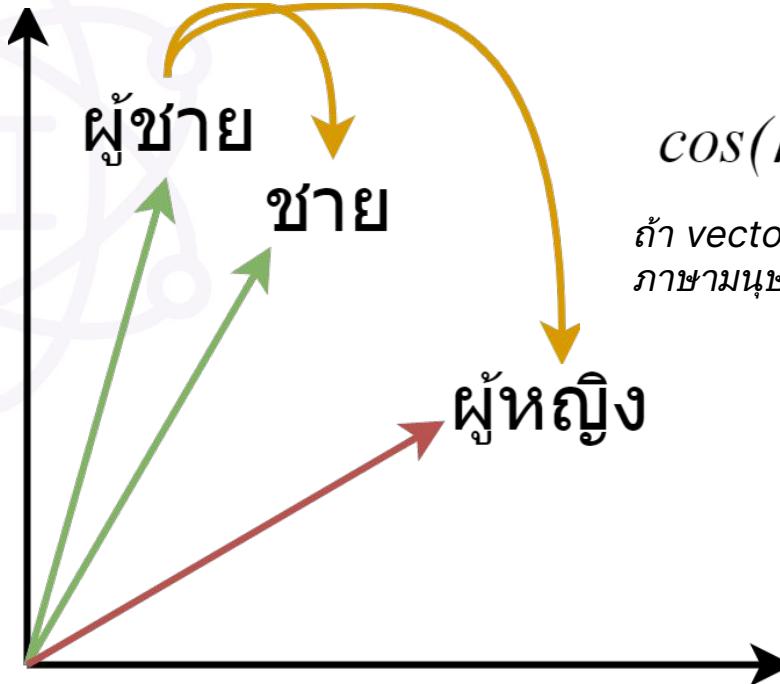
## ตัดคำ ตัดพยางค์ และทำ Bag-of-Word

ข้อดี	ข้อเสีย
<ul style="list-style-type: none"><li>ไม่ยุ่งยาก ทำได้ง่ายสุด</li><li>ประสิทธิภาพยังดีมาก เมื่นอยุคปัจจุบัน</li></ul>	<ul style="list-style-type: none"><li>ต้องพึงการตัดคำมากไป</li><li>BoW ไม่มีลำดับของคำมาเกี่ยว</li><li>ไม่รองรับคำที่ไม่เคยเห็น</li></ul>

## แปลงข้อความเป็น Vector (Discrete)



## แปลงข้อความเป็น Vector



*Cosine similarity ( $\cos(\bullet)$ )*

$$\cos(\text{ผู้ชาย}, \text{ชาย}) > \cos(\text{ผู้ชาย}, \text{ผู้หญิง})$$

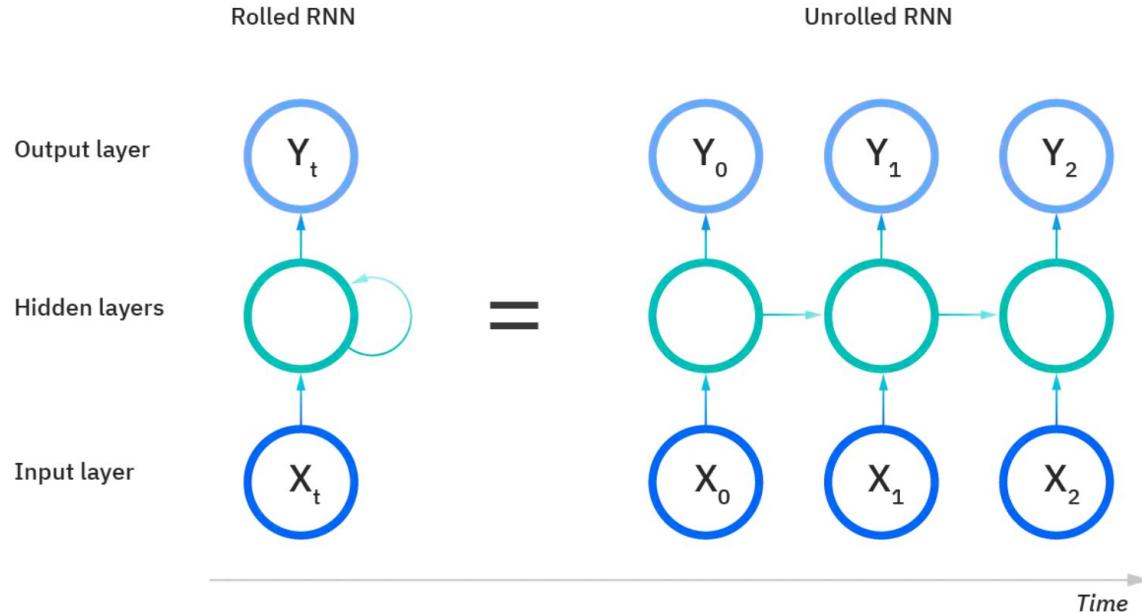
ถ้า vector ของสามารถแปลงความหมายได้ดี = คอมพิวเตอร์เข้าใจภาษาคนนุชย์ได้ดี

# แปลงข้อความเป็น Vector

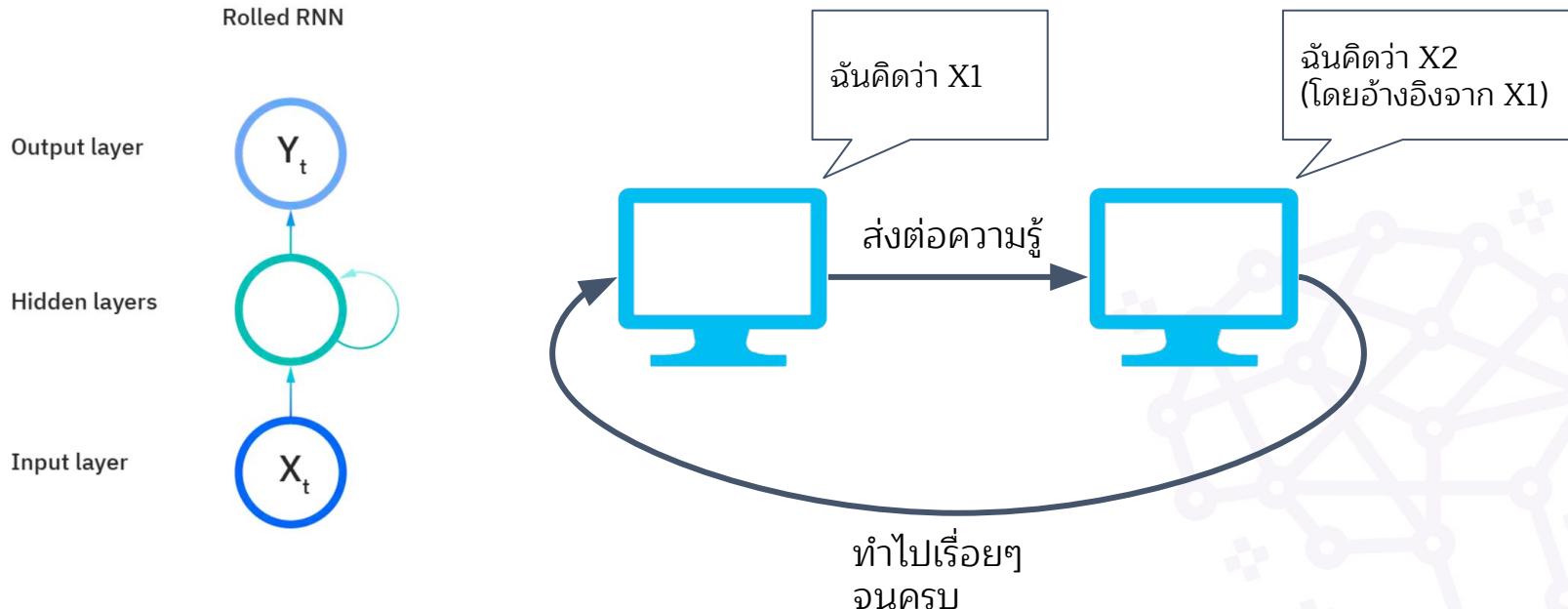
ข้อดี	ข้อเสีย
<ul style="list-style-type: none"><li>มีประสิทธิภาพสูงในงานที่ต้องการความซับซ้อน</li><li>รองรับคำที่ไม่เคยเห็น</li></ul>	<ul style="list-style-type: none"><li>การเทรนโมเดลมีความซับซ้อนกว่า BoW</li></ul>

# แล้วสร้าง Vector ออกมายังไง?

# ຢຸດທອງນອງ “RNN”



# ຢຸດທອງຂອງ “RNN”



# ຢຸຄທອງນອງ “RNN”

[cstorm125 / thai2fit](#) Public

Code Issues 1 Pull requests Actions Projects Wiki Security

master 1 branch 1 tag Go to file Add file Code

cstorm125 add citation details 7e58550 on Jan 9, 2021 51 commits

- images add submission image 3 years ago
- thwiki\_lm pycon 2019 3 years ago
- wongnai\_cls Add USE benchmark 2 years ago
- .gitattributes Ifs 4 years ago
- .gitignore wongnai benchmarks all done 3 years ago
- LICENSE Initial commit 4 years ago
- README.md add citation details 14 months ago

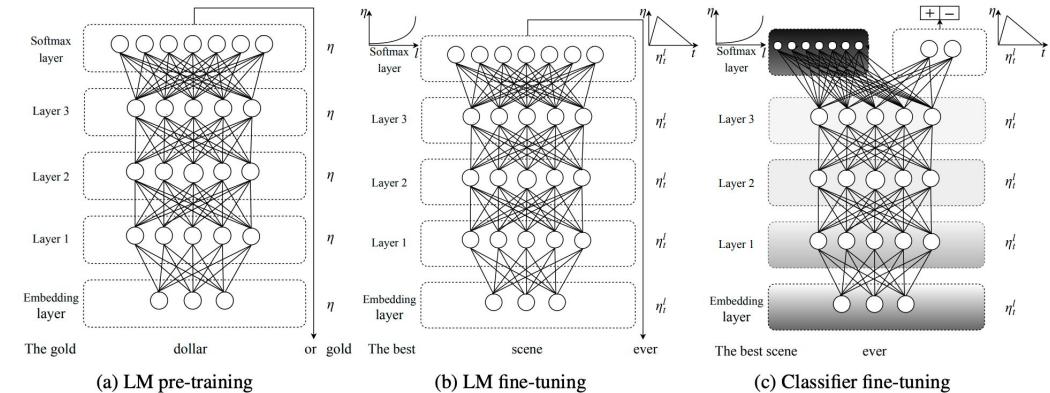
README.md

## thai2fit (formerly thai2vec)

ULMFit Language Modeling, Text Feature Extraction and Text Classification in Thai Language. Created as part of [pyThaiNLP](#) with ULMFit implementation from [fast.ai](#)

Models and word embeddings can also be downloaded via [Dropbox](#).

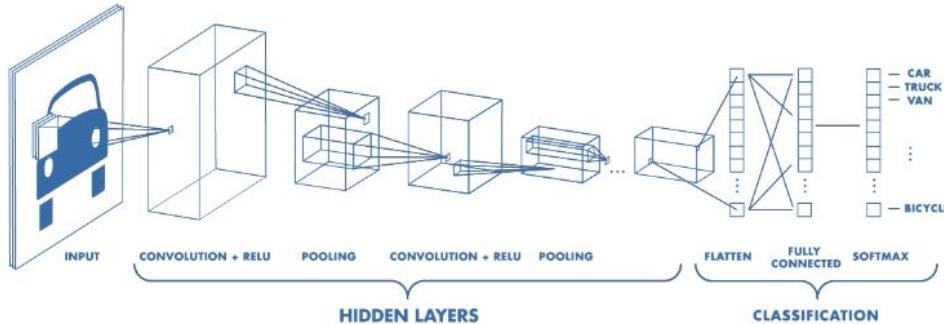
We pretrained a language model with 60,005 embeddings on [Thai Wikipedia Dump](#) (perplexity of 28.71067) and text classification (micro-averaged F-1 score of 0.60322 on 5-label classification problem. Benchmarked to 0.5109 by [fastText](#) and 0.4976 by LinearSVC on [Wongnai Challenge: Review Rating Prediction](#). The language model can also be used to extract text features for other downstream tasks.



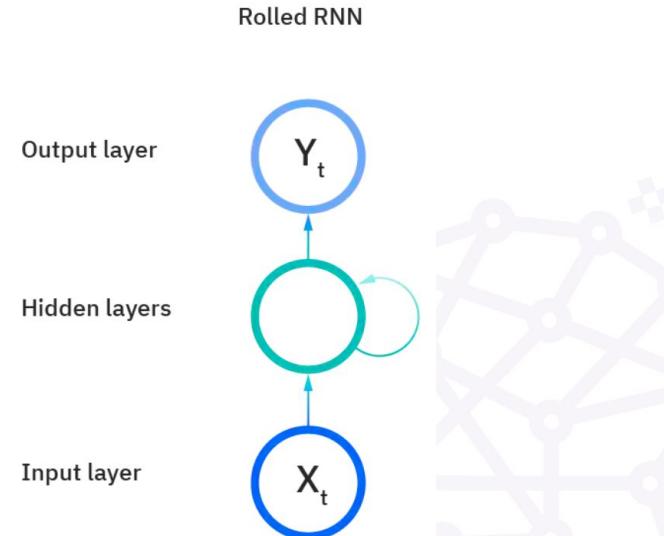
## ข้อดีของ “RNN”

ข้อดี	ข้อเสีย
<ul style="list-style-type: none"><li>มีประสิทธิภาพสูงในงานที่ต้องการความซับซ้อน</li><li>ลำดับของคำ มีความสำคัญ</li></ul>	<ul style="list-style-type: none"><li>ใช้เวลาในการเทรนโน้มเดลเป็นเวลานาน</li><li>ยิ่งวน ยิ่งลืมของก่อนหน้า หรือยิ่งซับสน</li></ul>

# บุคคลของ “BERT”



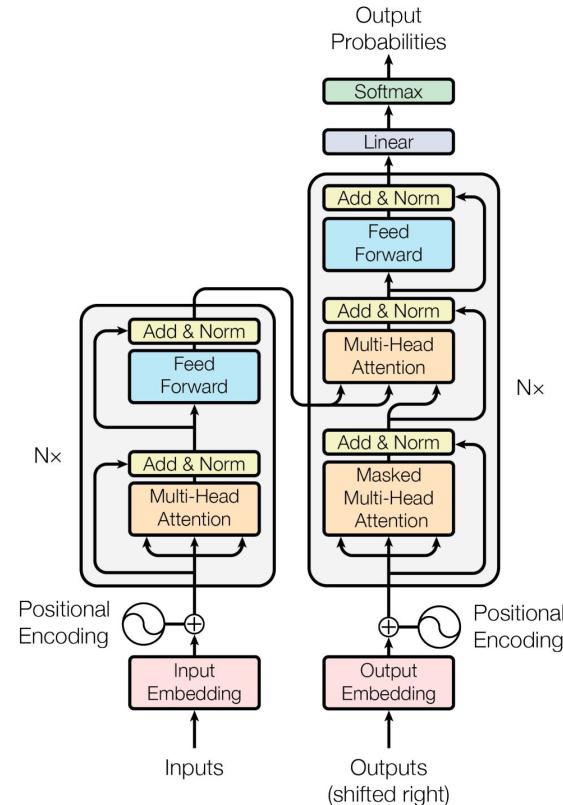
ไม่นานเมื่อ RNN แต่หมายความกับ classification ที่ไม่ใช้ order



เหตุนนาน แต่หมายความกับ classification ที่ order มีผล

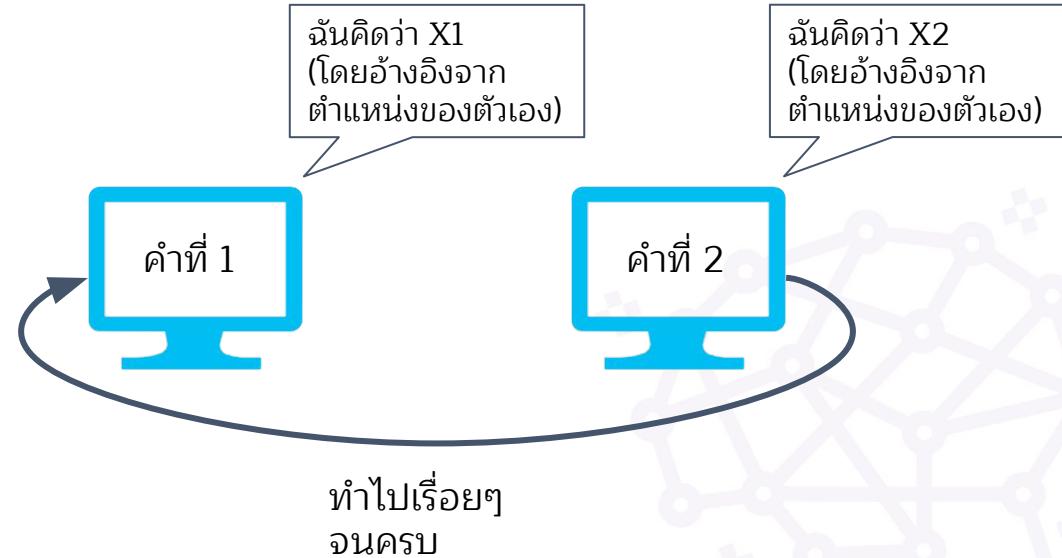
# ยุคทองของ “BERT”

- Transformer: รวมข้อดีของ CNN และ RNN
  - คำนวณ output แบบไม่สนใจ order
  - แต่มีการกำหนดตำแหน่งที่ input แทน

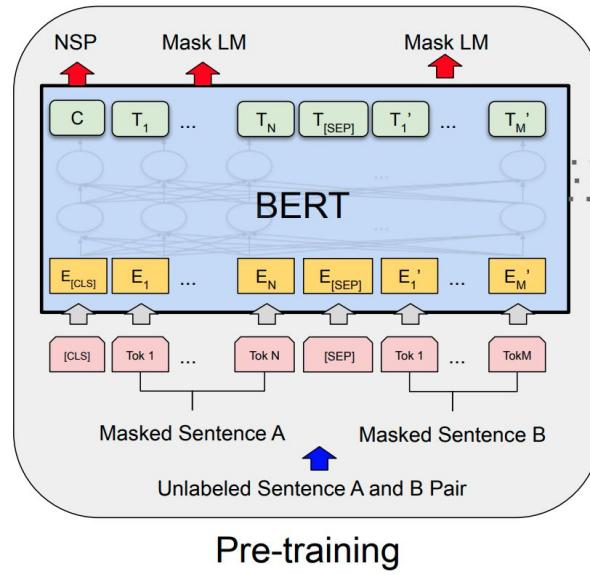


## ยุคทองของ “BERT”

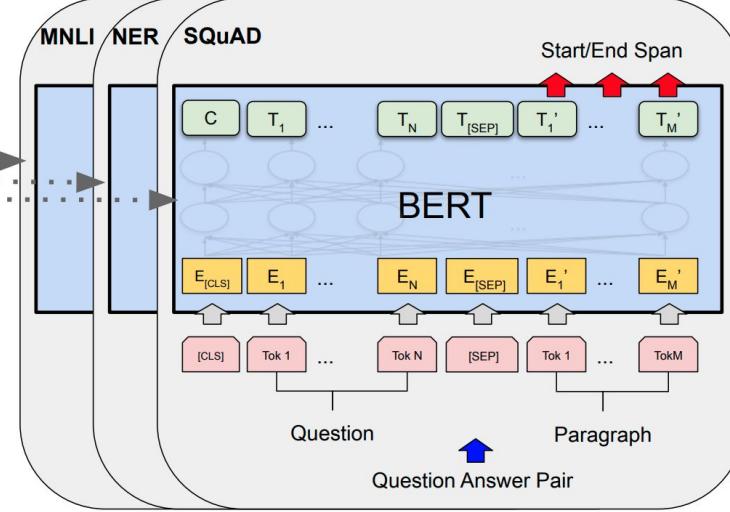
อธิบาย Transformer แบบเข้าใจง่ายๆ



# ຢຸຄທອງບອນ “BERT”



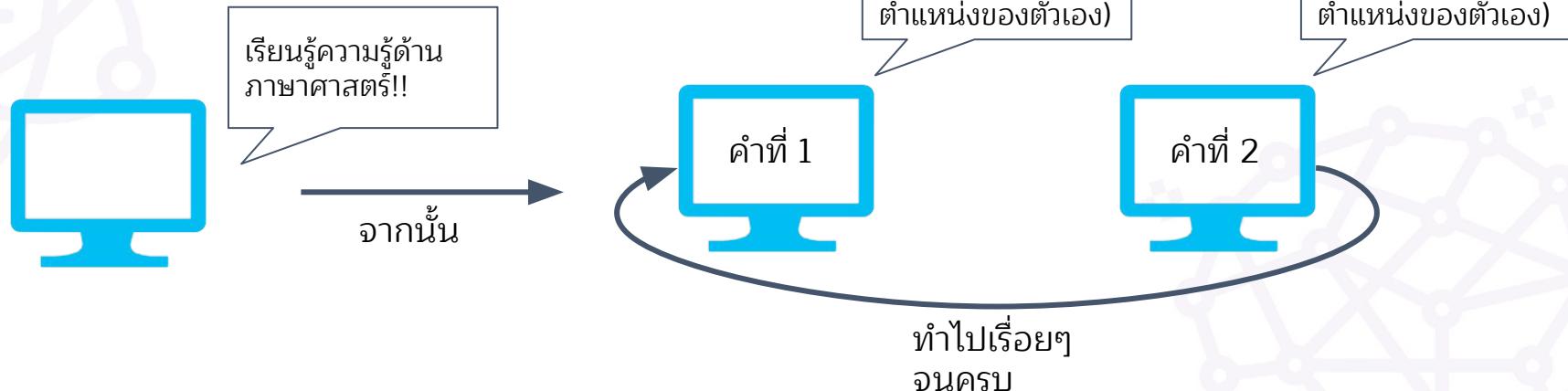
Pre-training



Fine-Tuning

## บุคคลของ “BERT”

### อธิบาย BERT แบบเข้าใจง่ายๆ



# บุคคลองของ “BERT”



VISTEC-depa AI Research Institute of Thailand

Jan 24, 2021 · 5 min read

## WangchanBERTa โมเดลประมวลผลภาษาไทยที่ใหญ่ และก้าวหน้าที่สุด ในขณะนี้

เปิดให้ทุกคนใช้ฟรีโดย AIResearch.in.th และ VISTEC ภายใต้ลิขสิทธิ์อนุญาต CC-BY-SA 4.0



Image by Phannisa Nirattiwongsakorn

## การจำแนกข้อความภาษาไทย ตั้งแต่ BoW ⇒ BERT

Let's code!

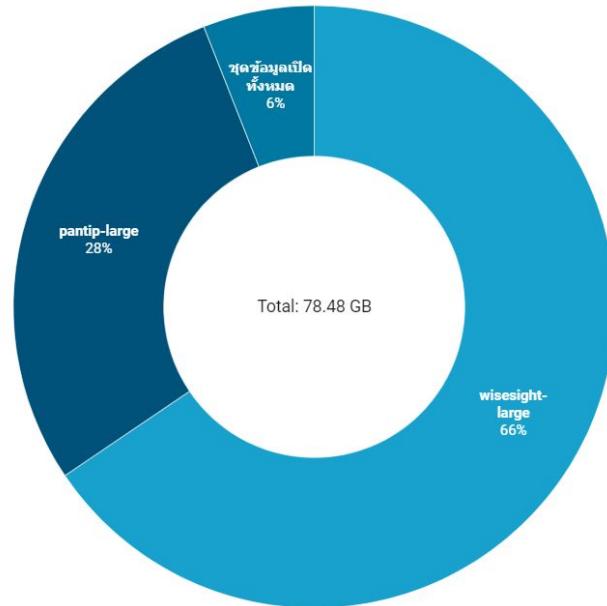


# NLP ภาษาไทยไปถึงไหนแล้ว?

# โมเดลและคลังข้อมูลสารสนเทศ

## ขนาดชุดข้อมูลทั้งหมด (78.48GB)

wisesight-large    pantip-large    ชุดข้อมูลเปิดทั่วโลก



## เทรนโมเดลที่ใหญ่ที่สุดในประวัติศาสตร์ NLP ไทย

เราเลือกใช้สถาปัตยกรรม RoBERTa ที่สามารถเทรนได้ด้วยงาน masked language model (MLM) เท่านั้น ไม่เหมือน BERT ที่ต้องทำการทำนายประโยคต่อท้าย (sentence entailment) ด้วย แต่การจะเทรนโมเดลที่ใหญ่ที่สุดเท่าที่เคยถูกเทรนบนภาษาไทยภาษาเดียวให้ดีนั้น มันไม่ได้ง่ายขนาดแค่กดรันสคริปท์ตาม tutorial ของ HuggingFace (แต่มันก็เป็น tutorial ที่ดีมาก เราแนะนำให้ไปลองอ่าน)

ปัญหาสำคัญคือ ความแตกต่างในเชิงทรัพยากรการคำนวณ ในขณะที่เราเทรนโมเดลด้วย Nvidia DGX-1 (ราคาราด 129,000 ดอลลาร์สหรัฐฯหรือประมาณเครื่องละ 4 ล้านบาท) ซึ่งประกอบด้วย GPU รุ่น Nvidia Tesla V100 ขนาด 32GB จำนวน 8 หน่วย ใช้เวลาประมาณ 125 วัน 19 ชั่วโมงต่อหนึ่งรอบการเทรนให้ครบ

# โมเดลและคลังข้อมูลสารสนเทศ

README.md



## PyThaiNLP: Thai Natural Language Processing in Python

[pypi v3.0.5](#) [python 3.7](#) [License Apache 2.0](#) [downloads/month 57k](#) [Unit test and code coverage passing](#)  
[coverage 97%](#) [code quality A](#) [license scan failing](#) [Launch Quick Start Guide on Google Colab](#)  
[DOI 10.5281/zenodo.6075269](#) [matrix join chat](#)

PyThaiNLP is a Python package for text processing and linguistic analysis, similar to [NLTK](#) with focus on Thai language.

PyThaiNLP เป็นไลบรารีภาษาไพทอนสำหรับประมวลผลภาษาธรรมชาติ คล้ายกับ NLTK โดยเน้นภาษาไทย รายละเอียดภาษาไทยได้ที่ [README\\_TH.MD](#)

### News

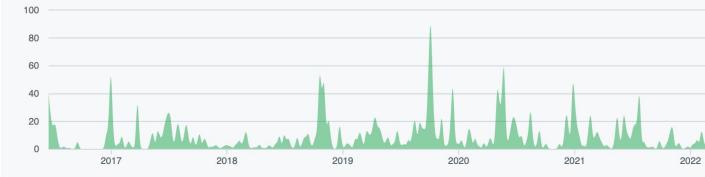
Now, You can contact or ask any questions you encounter with the PyThaiNLP team.  
[matrix join chat](#)

Version	Description	Status
3.0	Stable	<a href="#">Change Log</a>
dev	Release Candidate for 3.1	<a href="#">Change Log</a>

Jun 19, 2016 – Mar 6, 2022

Contributions: Commits ▾

Contributions to dev, excluding merge commits and bot accounts



# โมเดลและคลังข้อมูลสาธารณะ



ชุดข้อมูลและโมเดล ติดต่อเรา

## ชุดข้อมูลและโมเดลที่น่าสนใจ

### Thai Speech Emotion Dataset

ชุดข้อมูลจำแนกภาระณ์จากเสียงพูดภาษาไทย

26 Mar 2021

dataset  v1.0

### WangchanBERTa: Pre-trained Thai Language Model

โมเดลภาษาสำหรับงานประมวลผล และการเข้าใจภาษาไทย

3 Mar 2021

model  v1.0

### English-Thai Machine Translation Models

โมเดลแปลภาษา อังกฤษ-ไทย จากชุดข้อมูลกว่า 1 ล้าน คู่ประโยค

23 Jun 2020

model  demo  v1.0

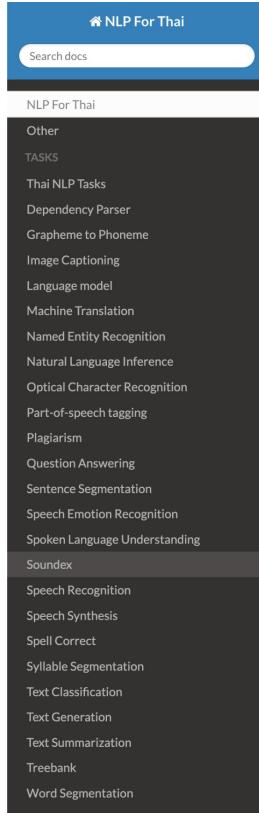
### English-Thai Machine Translation Dataset

ชุดข้อมูลคุณภาพโดยคุณภาพภาษาอังกฤษ-ไทย กว่า 1 ล้านคู่ประโยค

23 Jun 2020

dataset  v1.0

# ໂນໂດລແລະຄລັງຂ້ອມມຸລສານາຮນະ



Docs » NLP For Thai

## NLP For Thai

It's Thai NLP homepage. All is Open Source.

Website: [NLPForThai.com](https://NLPForThai.com) maintained by PyThaiNLP

### Menu

- [Tasks](#)
- [Other](#)

### Contributors



Thanks all the [contributors](#). (Image made with [contributors-img](#))

### How to Contribute

You can fork and send your pull request at <https://github.com/PyThaiNLP/nlpforthai.com>

We build Thai NLP.

PyThaiNLP

Next 

Built with [MkDocs](#) using a theme provided by [Read the Docs](#).

<https://nlpforthai.com/>

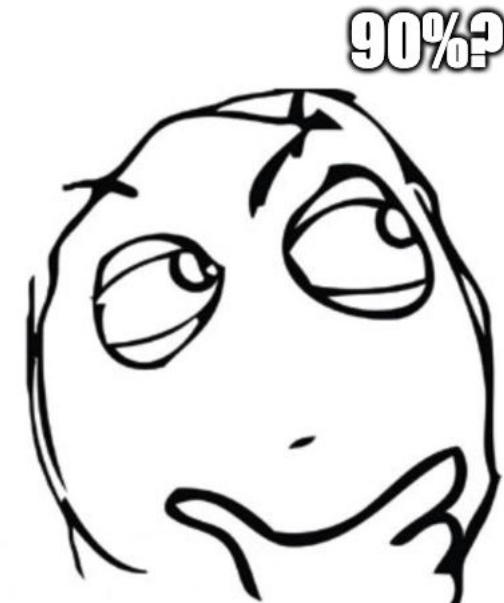
อะไรที่ภาษาไทยยังทำไม่ได้?



# อะไรที่ภาษาไทยยังทำไม่ได้?

## ตัดคำ

Method	WS160		TNHC		LST20		VISTEC	
	Char	Word	Char	Word	Char	Word	Char	Word
DC	93.47	84.03	89.48	75.40	94.60	87.15	92.77	81.78
AC	93.50	84.04	88.82	73.71	95.24	87.21	91.47	79.31
TL-DC	96.30	90.60	95.43	88.60	98.63	96.30	96.78	90.99
TL-AC	94.10	85.00	90.57	77.54	98.04	94.77	95.47	89.27
SE-DC	95.20	86.90	95.20	84.10	94.96	87.72	94.76	86.33
SE-AC	94.50	85.60	93.70	83.90	96.30	89.87	93.86	84.43
DSE-DC	<b>96.67</b>	<b>91.51</b>	<b>95.71</b>	<b>89.14</b>	<b>99.01</b>	<b>97.33</b>	<b>97.36</b>	<b>92.91</b>
DSE-AC	94.57	86.24	95.51	88.52	98.46	95.79	97.31	92.78



# อะไรที่ภาษาไทยยังทำไม่ได้?

## การจำแนกข้อความ

	โมเดล-ชุดข้อมูล	Wisesight Sentiment	Wongnai Reviews	Generated Reviews EN-TH	Prachathai67k
1	NBSVM	72.03	58.38	59.68	66.77
2	ULMFit	70.95	61.79	64.33	66.21
3	XLMR	73.57	62.57	<b>64.91</b>	68.18
4	mBERT	70.05	47.99	62.14	66.47
5	WanchanBERTa	<b>76.19</b>	<b>63.05</b>	64.66	<b>69.78</b>

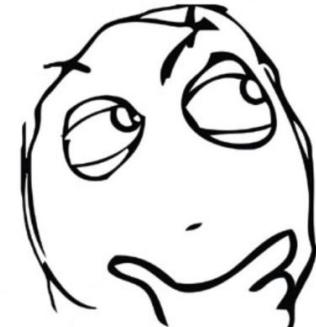


# อะไรที่ภาษาไทยยังทำไม่ได้?

## เครื่องแปลภาษา

	Google	AI for Thai	Our Baseline (SCB_1M)	Our Baseline (MT_OPUS)	Our Baseline (SCB_1M+MT_OPUS)
<i>Thai → English IWSLT 2015</i>					
SacreBLEU (case-sensitive)	14.19 (46.7/19.9/10.0/5.1)	*	17.2 (50.7/23.1/12.1/6.6)	28.1 (60.8/35.6/23.1/15.)	<b>28.3</b> (60.8/35.6/22.9/15.1)
SacreBLEU (case-insensitive)	17.64 (53.8/24.5/12.7/6.8)	*	17.93 (52.4/24.0/12.7/7.0)	28.7 (62.0/36.3/23.7/16.)	<b>29.0</b> (62.0/36.4/23.5/15.6)
<i>English → Thai IWSLT 2015</i>					
BLEU4	15.36 (51.0/23.8/12.0/6.2)	6.14 (36.1/11.7/4.3/1.7)	12.95 (45.5/19.5/9.0/4.3)	17.24 (52.0/26.3/14.4/8.1)	<b>17.77</b> (52.2/26.7/14.8/8.5)

ดีกว่า GOOGLE?



# อะไรที่ภาษาไทยยังทำไม่ได้?

ทำตัดคำไม่ได้



NLP ในยุคต้น 2000

มีโมเดลเทพอหิ  
จะไม่ต้องแดร์ผล  
การตัดคำ



NLP ในยุค 2020

# เริ่มทำ NLP เริ่มยังไงดี?

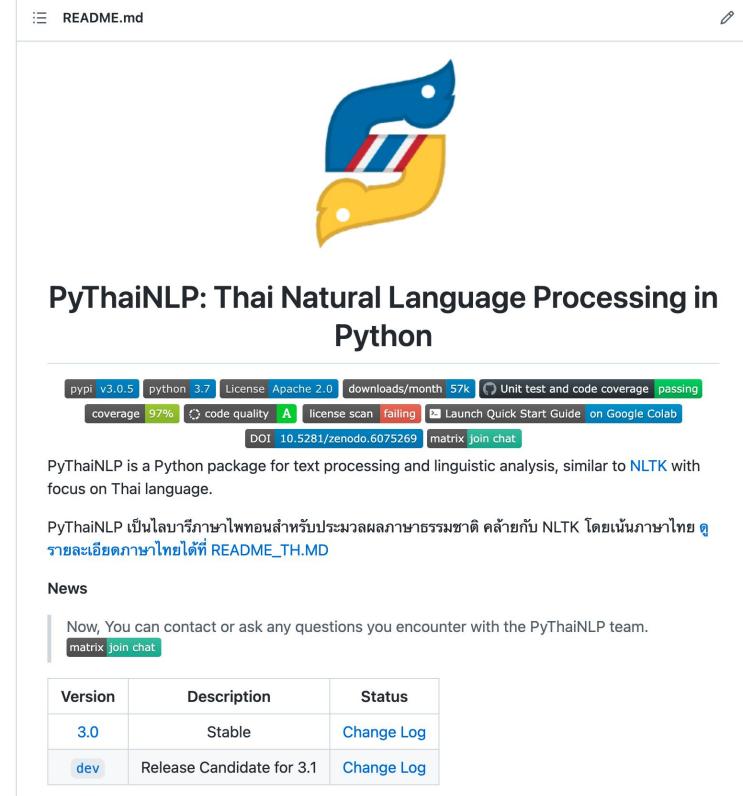


The screenshot shows the GitHub page for the `ThaiNLP` repository. The main title is "Thai Natural Language Processing". Below it, the subtitle is "กลุ่มคนทำ NLP ภาษาไทย". The repository has 1.2 thousand stars and 1.2 thousand forks. A button at the bottom right says "+ เข้าร่วม".

## Thai Natural Language Processing

กลุ่มสาธารณะ · สมาชิก 1.2 หมื่น คน

 ได้เข้าร่วม ▾ + เข้าร่วม ▾



The screenshot shows the GitHub page for the `PyThaiNLP` repository. The title is "PyThaiNLP: Thai Natural Language Processing in Python". It features a logo of a stylized bird. Key statistics include 57k downloads/month, 97% coverage, and failing code quality. The repository is licensed under Apache 2.0. A note states: "PyThaiNLP เป็นไลบรารีภาษาไพทอนสำหรับประมวลผลภาษาธรรมชาติ คล้ายกับ NLTK โดยเน้นภาษาไทย รายละเอียดภาษาไทยได้ที่ README\_TH.MD". The "News" section contains a message about contacting the team via Matrix. A table at the bottom lists the version, description, and status.

README.md



## PyThaiNLP: Thai Natural Language Processing in Python

pypi v3.0.5 python 3.7 License Apache 2.0 downloads/month 57k Unit test and code coverage passing  
coverage 97% code quality A license scan failing Launch Quick Start Guide on Google Colab  
DOI 10.5281/zenodo.6075269 matrix join chat

PyThaiNLP is a Python package for text processing and linguistic analysis, similar to [NLTK](#) with focus on Thai language.

PyThaiNLP เป็นไลบรารีภาษาไพทอนสำหรับประมวลผลภาษาธรรมชาติ คล้ายกับ NLTK โดยเน้นภาษาไทย รายละเอียดภาษาไทยได้ที่ [README\\_TH.MD](#)

### News

Now, You can contact or ask any questions you encounter with the PyThaiNLP team.  
[matrix join chat](#)

Version	Description	Status
3.0	Stable	<a href="#">Change Log</a>
dev	Release Candidate for 3.1	<a href="#">Change Log</a>

# คอร์สแนะนำ



Natural Language Processing

วิดีโอ 15 รายการ • ภาระ 25,302 ครั้ง • อัปเดตล่าสุด เมื่อ 9 เม.ย. 2021

Course github: [https://github.com/ekapolc/nlp\\_course](https://github.com/ekapolc/nlp_course)

Ekapol Chuangsuwanich

ติดตามแล้ว

...

1 2110594 NLP L1 Introduction  
Ekapol Chuangsuwanich 2:03:19

2 2110594 NLP L2 Tokenization: Intro to Neural Networks  
Ekapol Chuangsuwanich 2:36:05

3 2110594 NLP L3 Language Modeling  
Ekapol Chuangsuwanich 1:25:49

4 2110594 NLP L4 Word representation  
Ekapol Chuangsuwanich 2:20:28

5 2110594 NLP L5 PoS Tagging  
Ekapol Chuangsuwanich 2:26:50

# คอร์สแนะนำ

## Text Classification

โจทย์ส่วนใหญ่ของ NLP นั้นสามารถแก้ได้ด้วยการสร้างเครื่องจำแนกประเภทข้อความ (Text Classifier) ซึ่งมี machine learning algorithm เป็นแกนหลัก (สามารถศึกษาพื้นฐานของ machine learning จาก module **sentiment analysis**) ปัจจุบันมีคนหันมาใช้ Neural Network กันมากขึ้น โดยใช้พื้นฐานของความหมายของคำที่กำหนดโดยบริบทและการใช้ภาษา (Distributed semantic model หรือ distributional semantic model) ในคลังข้อมูลเป็นหลัก



- Part of speech tagging : แบ่งชนิดของคำ
- Input: คำแต่ละคำ
- Output: {Noun, Adj, Verb, Adv, Preposition, ...}

## TEXT CLASSIFICATION



### Video List:

1. Text Classification คืออะไร
2. Logistic Regression แบบไม่ต้องเขียนสมการ
3. Crossentropy Loss ของ Logistic Regression
4. วิธีการฝึกโมเดลด้วย scikit-learn
5. Sparse features and sparse matrix
6. Data structure: numpy array and dense matrix
7. Data structure: Sparse Matrix

เนื้อหาสำหรับอ่านและอ้างอิง

- Slides จากวิธีโอเร็ง Text classification
- Chapter 4 จาก Jurafsky
- Chapter 5 จาก Jurafsky



# THANK YOU

