

# Natural Language Processing

Ali Akbar Septiandri

Universitas Al-Azhar Indonesia

*aliakbars@live.com*

May 15, 2017

## 1 Natural Language Processing

- Pendahuluan
- Representasi

## 2 NLTK

- Pengenalan
- Demo dan Alternatif

- 1 Bird, S., Edward L. & Klein, E. (2009). Natural Language Processing with Python. OReilly Media Inc.
- 2 Jurafsky, D. & Martin, J. H. (2014). Speech and Language Processing (Vol. 3). Pearson.

# Natural Language Processing

# Apa Itu NLP?

Salah satu ilmu multidisiplin yang berfokus pada interaksi manusia dan komputer melalui bahasa alami manusia. Beberapa hal yang dibahas di dalamnya antara lain:

- Part-of-Speech (POS) tagging
- Parsing
- Stemming
- Machine translation
- Named entity recognition (NER)
- Question answering
- Sentiment analysis
- Automatic summarisation
- Speech recognition
- Text-to-speech

# Kategori Tugas-tugas NLP

- Syntax
  - ▶ Part-of-Speech (POS) tagging
  - ▶ Parsing
  - ▶ Stemming
- Semantics
  - ▶ Machine translation
  - ▶ Named entity recognition (NER)
  - ▶ Question answering
  - ▶ Sentiment analysis
- Discourse
  - ▶ Automatic summarisation
- Speech
  - ▶ Speech recognition
  - ▶ Text-to-speech



- NLP juga dikenal dengan nama *computational linguistics*, karena mencoba merepresentasikan makna dari kata, frasa, kalimat, dan dokumen melalui **distribusinya**



- NLP juga dikenal dengan nama *computational linguistics*, karena mencoba merepresentasikan makna dari kata, frasa, kalimat, dan dokumen melalui **distribusinya**
- Distribusi tersebut direpresentasikan dalam **vektor konteks**

- NLP juga dikenal dengan nama *computational linguistics*, karena mencoba merepresentasikan makna dari kata, frasa, kalimat, dan dokumen melalui **distribusinya**
- Distribusi tersebut direpresentasikan dalam **vektor konteks**
- “Dalam suatu dokumen, kata apa saja yang muncul bersamaan?”

- NLP juga dikenal dengan nama *computational linguistics*, karena mencoba merepresentasikan makna dari kata, frasa, kalimat, dan dokumen melalui **distribusinya**
- Distribusi tersebut direpresentasikan dalam **vektor konteks**
- “Dalam suatu dokumen, kata apa saja yang muncul bersamaan?”
- Begitu pula di level semantik → **Bag-of-Words (BoW) model**

- NLP juga dikenal dengan nama *computational linguistics*, karena mencoba merepresentasikan makna dari kata, frasa, kalimat, dan dokumen melalui **distribusinya**
- Distribusi tersebut direpresentasikan dalam **vektor konteks**
- “Dalam suatu dokumen, kata apa saja yang muncul bersamaan?”
- Begitu pula di level semantik → **Bag-of-Words (BoW) model**
- Bahkan, bisa sampai ke level **karakter!**

Dalam representasi ini, urutan atau letak dari kata tersebut tidak relevan

- D1 “send us your password”
- D2 “send us your review”
- D3 “review your password”
- D4 “review us”
- D5 “send your password”
- D6 “send us your account”

# Binary Bag-of-Words

Dalam representasi ini, urutan atau letak dari kata tersebut tidak relevan

<b>dokumen</b>	account	password	review	send	us	your
D1	0	1	0	1	1	1
D2	0	0	1	1	1	1
D3	0	1	1	0	0	1
D4	0	0	1	0	1	0
D5	0	1	0	1	0	1
D6	1	0	0	1	1	1

$$w_{t,d} = (1 + \log(tf_{t,d})) \log\left(\frac{N}{df_t}\right)$$

- $tf_{t,d}$  ... frekuensi kata  $t$  dalam dokumen  $d$ ,  $N$  ... jumlah dokumen,  $df_t$  ... jumlah dokumen yang mempunyai kata  $t$
- Kata yang sering muncul mungkin tidak penting, e.g. kata hubung
- Kata yang langka akan bernilai lebih – lihat posisi  $df_t$ !

# Menemukan Dokumen yang Mirip

- Dalam contoh minggu lalu, kita menggunakan **Euclidean distance**
- Untuk dokumen, jumlah kemunculan kata  *mungkin* tidak begitu penting
- Yang penting adalah keberadaan katanya → **cosine similarity**

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \cdot \|\mathbf{y}\|}$$



- Dengan ide yang serupa dan beberapa tambahan algoritma lainnya, e.g. *Latent Semantic Analysis* (LSA), kita bisa menggunakan kakas ini untuk **tes seperti TOEFL**

- Dengan ide yang serupa dan beberapa tambahan algoritma lainnya, e.g. *Latent Semantic Analysis* (LSA), kita bisa menggunakan kakas ini untuk **tes seperti TOEFL**
- LSA berhasil menjawab 64.4% soal

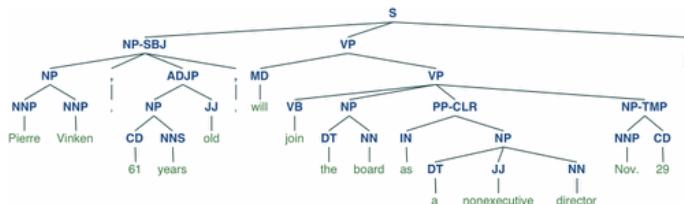
- Dengan ide yang serupa dan beberapa tambahan algoritma lainnya, e.g. *Latent Semantic Analysis* (LSA), kita bisa menggunakan kakas ini untuk **tes seperti TOEFL**
- LSA berhasil menjawab 64.4% soal
- Pengguna bahasa Inggris non-natif rata-rata berhasil menjawab 64.5% soal

- Dengan ide yang serupa dan beberapa tambahan algoritma lainnya, e.g. *Latent Semantic Analysis* (LSA), kita bisa menggunakan kakas ini untuk **tes seperti TOEFL**
- LSA berhasil menjawab 64.4% soal
- Pengguna bahasa Inggris non-natif rata-rata berhasil menjawab 64.5% soal
- *Cukup untuk masuk banyak universitas di US!*

# NLTK

“NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, **tokenization**, **stemming**, **tagging**, **parsing**, and **semantic reasoning**...”

# NER Tagging



Gambar: Hasil NER tagging dengan NLTK [NLTK Project, 2017]

Beberapa korpus dan model yang terkenal dari NLTK:

- Project Gutenberg Selections
- Penn Treebank
- SentiWordNet
- Stopwords Corpus
- Porter Stemmer



# Everything Data

Document Similarity using NLTK and Scikit-Learn

Beberapa alternatif untuk tugas-tugas spesifik:

- **spaCy**: Industrial-Strength Natural Language Processing in Python
- **gensim**: topic modelling for humans

# References



NLTK Project (2 January 2017)

Natural Language Toolkit

<http://www.nltk.org/>



Richard Socher (accessed on 15 May 2017)

CS224d: Deep Learning for Natural Language Processing

<http://cs224d.stanford.edu/>

Terima kasih