Dampak Penggunaan Anotasi Penamaan yang Berbeda Pada Kinerja NER

I Made Widi Arsa Ari Saputra[a1], I Wayan Supriana[a2]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana
Jalan Raya Kampus Udayana, Bukit Jimbaran, Kuta Selatan, Badung, Bali Indonesia
[1]widiarsa.sama@gmail.com
[2]wayan.supriana@unud.ac.id.com

Abstract

In developing the NER model, naming annotations are used as an important part of the training process. The impact of using different naming annotations on NER performance has been a concern in the research community. So, the writer wants to once again, test the impact of using different naming annotations using the spaCy library on English documents. Using 2 naming schemes namely BIO and IOBES, using the spaCy model to get 0.96 accuracy for BIO and 0.95 for IOBES.

Keywords: *NER, Person Entity, spaCy, BIO, IOBES, Named Entity Annotation*

1. Pendahuluan

Named Entity Recognition (NER) adalah salah satu tugas penting dalam pemrosesan bahasa alami (Natural Language Processing, NLP) yang bertujuan untuk mengidentifikasi entitas yang signifikan dalam teks, seperti orang, tempat, organisasi, tanggal, dan lain-lain. NER memiliki berbagai aplikasi, termasuk analisis teks, sistem tanya-jawab, dan ekstraksi informasi.

Dalam pengembangan model NER, anotasi penamaan digunakan sebagai bagian penting dalam proses pelatihan. Anotasi penamaan melibatkan memberikan label entitas yang benar kepada kata-kata dalam teks. Namun, masalah muncul ketika penggunaan anotasi penamaan yang berbeda-beda digunakan dalam dataset pelatihan, yang dapat berdampak pada kinerja model NER.

Dampak penggunaan anotasi penamaan yang berbeda pada kinerja NER telah menjadi perhatian dalam komunitas penelitian. Penelitian oleh Chen et al. [1] menunjukkan bahwa inkonsistensi dalam anotasi penamaan dapat menyebabkan penurunan signifikan dalam kinerja model NER. Mereka menemukan bahwa variasi dalam anotasi penamaan dapat menghasilkan ambiguitas dan kesulitan dalam melatih model secara efektif.

Selain itu, penelitian oleh Yang et al. [3] juga mengungkapkan bahwa perbedaan dalam anotasi penamaan dapat mempengaruhi konsistensi dan keandalan model NER. Ketika dataset pelatihan mengandung anotasi penamaan yang tidak konsisten, model cenderung menghasilkan hasil yang tidak dapat diandalkan dan mengalami kesulitan dalam mengenali entitas dengan benar.

Oleh karena itu, penting untuk mempertimbangkan penggunaan anotasi penamaan yang konsisten dalam pengembangan model NER. Dengan memastikan konsistensi dalam anotasi penamaan, dapat meningkatkan kinerja model dan menghasilkan hasil yang lebih akurat. Sehingga penulis ingin sekali lagi, menguji dampak penggunaan anotasi penamaan yang berbeda menggunakan model spaCy pada dokumen berbahasa inggris.

2. Metode Penelitian

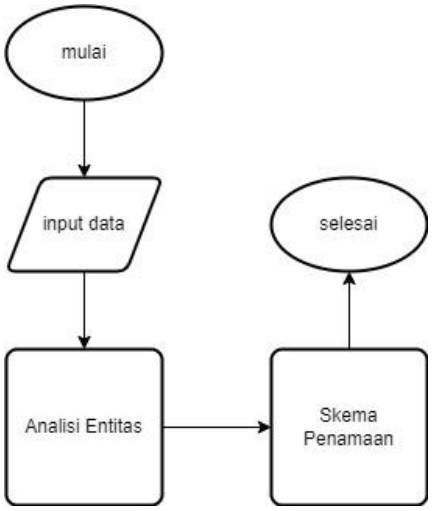
Dalam penelitian ini, akan menggunakan library spaCy untuk melakukan analisis

2.1 Pengumpulan Data

Pada penelitian ini, data yang digunakan adalah data teks yang didapatkan dari website bahasainggris.xyz berupa cerita pendek dari film harry potter menggunakan bahasa inggris. Data ini dibersihkan agar keseluruhan teks hanya mengandung bahasa inggris, sehingga proses identifikasi menggunakan library spaCy dapat lebih efisien.

2.2 Perancangan Sistem

Data input adalah teks mentah yang akan di analisis menggunakan objek nlp dengan model “en_core_web_sm” pada library spaCy. Skema penamaan akan digunakan uji coba terhadap 2 skema yaitu BIO dan IOBES.



Gambar 1. Rancangan Sistem

2.3 Named Entity Recognition

Named Entity Recognition (NER) adalah tugas dalam pemrosesan bahasa alami (Natural Language Processing, NLP) yang bertujuan untuk mengidentifikasi dan mengklasifikasikan entitas yang signifikan dalam teks, seperti orang, tempat, organisasi, tanggal, dan lain-lain. NER sangat penting dalam berbagai aplikasi NLP, termasuk pemahaman teks, analisis sentimen, ekstraksi informasi, dan sistem tanya-jawab[2].

Salah satu pendekatan yang umum digunakan dalam NER adalah dengan menggunakan anotasi penamaan dengan skema BIO (Begin, Inside, Outside) atau IOBES (Inside, Outside, Beginning, End, Single) [4]. Skema ini memungkinkan untuk memberikan label entitas pada setiap kata dalam teks.

Pada skema BIO, setiap kata dalam teks diberikan label yang menunjukkan apakah itu bagian dari entitas atau tidak. Label yang digunakan meliputi:

a. B-ENTITY: Menandakan awal entitas.
b. I-ENTITY: Menandakan kata di dalam entitas.
c. O: Menandakan kata di luar entitas.

Misalnya, dalam kalimat "John Mayer adalah seorang penyanyi yang terkenal", kata "John" diberi label "B-PERSON" (awal entitas orang), kata "Mayer" diberi label "I-PERSON" (di dalam entitas orang), dan kata-kata lain diberi label "O" (di luar entitas).

Sementara itu, skema IOBES memperluas skema BIO dengan menambahkan label untuk menunjukkan akhir entitas ("E") dan entitas tunggal ("S"). Label yang digunakan dalam skema IOBES meliputi:

a. I-ENTITY: Menandakan kata di dalam entitas.
b. O: Menandakan kata di luar entitas.
c. B-ENTITY: Menandakan awal entitas.
d. E-ENTITY: Menandakan akhir entitas.
e. S-ENTITY: Menandakan entitas tunggal.

Penerapan skema IOBES memungkinkan penanganan yang lebih baik untuk entitas yang melibatkan beberapa kata, seperti entitas yang terdiri dari dua kata atau lebih.

2.4 Library SpaCy

SpaCy adalah salah satu library pemrosesan bahasa alami (Natural Language Processing, NLP) yang populer dan kuat yang menyediakan berbagai fitur untuk analisis teks, termasuk Named Entity Recognition (NER). Library ini menawarkan berbagai objek dan fungsi yang dapat digunakan untuk melakukan NER dengan mudah dan efisien.

Objek utama dalam spaCy yang berhubungan dengan NER adalah Doc dan Token. Doc merepresentasikan dokumen teks yang akan diproses, sedangkan Token mewakili setiap kata dalam dokumen tersebut.

Berikut adalah beberapa objek dan fungsi yang relevan untuk melakukan NER menggunakan spaCy:

a. nlp: Objek ini adalah inti dari spaCy dan digunakan untuk memproses teks. Ia akan membuat objek Doc dari teks yang diberikan dan memasukkannya ke dalam aliran pemrosesan spaCy.
b. Doc: Objek ini merepresentasikan dokumen teks yang telah diproses. Ia berisi kumpulan Token yang mewakili setiap kata dalam teks.
c. Token: Objek ini mewakili setiap kata dalam dokumen. Ia menyimpan informasi seperti teks kata, posisi dalam dokumen, informasi morfologis, serta label entitas yang dikenali.
d. Token.ent_type_: Atribut ini menyimpan label entitas yang dikenali untuk setiap Token. Jika sebuah Token bukan bagian dari entitas, nilai atribut ini adalah '' (string kosong).
e. Token.ent_iob_: Atribut ini menyimpan tag IOB (Inside, Outside, Beginning) untuk setiap Token. Nilai atribut ini dapat berupa 'I' (Inside) jika Token berada di dalam entitas, 'O' (Outside) jika Token bukan bagian dari entitas, atau 'B' (Beginning) jika Token adalah awal entitas.

3. Hasil dan Pembahasan

3.1 Source Code untuk BIO

Table 1. Kode Program BIO

```
import spacy
import os
os.system("cls")

def analyze_entities(text):
    nlp = spacy.load("en_core_web_sm")
    doc = nlp(text)

    entities = []
```

```
    current_entity = []
    current_label = None

    for token in doc:
        if token.ent_type_ == "PERSON":
            if current_label == "B-PERSON":
                current_entity.append(token.text)
                current_label = "I-PERSON"
                entities.append((current_entity, current_label))
            else:
                if current_entity:
                    entities.append((current_entity, current_label))
                current_entity = [token.text]
                current_label = "B-PERSON"
                entities.append((current_entity, current_label))
        else:
            if current_entity:
                entities.append((current_entity, current_label))
                current_entity = []
                current_label = None

            if token.text.strip():
                entities.append(([token.text], "O"))

    if current_entity:
        entities.append((current_entity, current_label))

    return entities

text = "Harry Potter is a orphanage young boy who has lost parent since he was very little baby. Harry was born to a magician father and mother. They are not ordinary human being they can do a trick and spell. Harry lives together with Dursley's Family; uncaring Aunt Petunia, loathsome Uncle Vernon, and spoiled cousin Dudley. The Drusley family barely tolerate Harry, his aunt really don't care about him, his fatty cousin Dudley always bullies him everytime he stay at home. They were living in suburb of little Whinging, Surrey, London. Young Harry Potter never feel loved by parents, his private room at home was small cellar under the stair where we normally put cleaning stuff such us moob, swab, broom etc. He was Happy even he didn't grow up in very loving family. One day Harry is astonished to receive a letter addressed to him in the cupboar under the stairs (where he sleeps). Unfortunatelly before he can open the letter, Uncle Vernon takes it and tear them all up, since that day the same letter come to Harry, but he never get any chance to open and read it even once. One day when the Drusley Family go vacation to miserable shack small island on Harry's 11th birthday a giant called Hagrid arrives to their home and reveals that Harry is a wizard an he has been accepted at the Hogwarts School of Withcraft and Wizardry"

entities = analyze_entities(text)

for entity, label in entities:
    for i, word in enumerate(entity):
        print(f"{word}\t{label}")


with open("bio.txt", "w", encoding="utf-8") as f:
    for entity, label in entities:
        for i, word in enumerate(entity):
            text = f.write(f"{word}\t{label}\n")
```

3.2 Source Code untuk IOBES

Table 2. Kode Program IOBES

```
import spacy

def analyze_entities(text):
    nlp = spacy.load("en_core_web_sm")
    doc = nlp(text)

    entities = []
    current_entity = []
    current_label = None

    for token in doc:
        if token.ent_type_ == "PERSON":
            if current_entity:
                if len(current_entity) == 1:
                    entities.append((current_entity[0], "S-PERSON"))
                else:
                    entities.append((current_entity[0], "B-PERSON"))
                    for i in range(1, len(current_entity) - 1):
                        entities.append((current_entity[i], "I-PERSON"))
                    entities.append((current_entity[-1], "E-PERSON"))
                current_entity = []
                current_label = None

            current_entity.append(token.text)
            current_label = "B-PERSON"
        else:
            if current_entity:
                if len(current_entity) == 1:
                    entities.append((current_entity[0], "S-PERSON"))
                else:
                    entities.append((current_entity[0], "B-PERSON"))
                    for i in range(1, len(current_entity) - 1):
                        entities.append((current_entity[i], "I-PERSON"))
                    entities.append((current_entity[-1], "E-PERSON"))
                current_entity = []
                current_label = None

            if token.text.strip():
                entities.append((token.text, "O"))

    if current_entity:
        if len(current_entity) == 1:
            entities.append((current_entity[0], "S-PERSON"))
        else:
            entities.append((current_entity[0], "B-PERSON"))
            for i in range(1, len(current_entity) - 1):
                entities.append((current_entity[i], "I-PERSON"))
            entities.append((current_entity[-1], "E-PERSON"))

    return entities
```

```
text = "Harry Potter is a orphanage young boy who has lost parent since he was very little baby. Harry was born to a magician father and mother. They are not ordinary human being they can do a trick and spell. Harry lives together with Dursley's Family; uncaring Aunt Petunia, loathsome Uncle Vernon, and spoiled cousin Dudley. The Drusley family barely tolerate Harry, his aunt really don't care about him, his fatty cousin Dudley always bullies him everytime he stay at home. They were living in suburb of little Whinging, Surrey, London. Young Harry Potter never feel loved by parents, his private room at home was small cellar under the stair where we normally put cleaning stuff such us moob, swab, broom etc. He was Happy even he didn't grow up in very loving family. One day Harry is astonished to receive a letter addressed to him in the cupboar under the stairs (where he sleeps). Unfortunatelly before he can open the letter, Uncle Vernon takes it and tear them all up, since that day the same letter come to Harry, but he never get any chance to open and read it even once. One day when the Drusley Family go vacation to miserable shack small island on Harry's 11th birthday a giant called Hagrid arrives to their home and reveals that Harry is a wizard an he has been accepted at the Hogwarts School of Withcraft and Wizardry"

entities = analyze_entities(text)

for entity, label in entities:
    print(f"{entity}\t{label}")
with open("IOBES.txt", "w", encoding="utf-8") as f:
    for entity, label in entities:
        text = f.write(f"{entity}\t{label}\n")
```

3.3 Analisis Hasil

Hasil program menggunakan library spaCy sebagai berikut. Hasil dari pengenalan entitas disajikan dalam bentuk tabel

Table 3.Tabel Akurasi

Skema Penamaan	Label Benar	Akurasi
BIO	256	0.96
IOBES	253	0.95

Dari data input didapatkan total 266 entitas yang di recognisi. Skema penamaan BIO berhasil merekognisi sejumlah 256 entitas secara tepat. Sedangkan skema penamaan IOBES berhasil merekognisi sejumlah 253 entitas secara tepat.

4. Kesimpulan

Penggunaan skema penamaan yang berbeda berpengaruh kepada akurasi dari Named Entity Recognition pada library spaCy. Skema penamaan BIO mendapat akurasi tertinggi denga, akurasi 0.96. Library spaCy adalah model yang telah terlatih sebelumnya, melakukan re-trained pada model sesuai dengan kondisi teks akan berpotensi menaikkan akurasi dari model.

Daftar Pustaka

[1] S. Haryati, A. Sudarsono, and E. Suryana, "Implementasi Data Mining Untuk [1] Chen, D., Manning, C. D. & Jurafsky, D. (2019). Simple BERT Models for Relation Extraction and Semantic Role Labeling. Proceedings of the 2019 Conference of the Association for Computational Linguistics (ACL), 3543-3551.

[2] Nadeau, D., & Sekine, S. (2007). A survey of named entity recognition and classification. Lingvisticae Investigationes, 30(1), 3-26.

[3] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A. & Hovy, E. (2020). NERDS: Neural Named

Entity Recognition with Dual-Level Selection. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 5104-5113.

[4] Sang, E. F., & Buchholz, S. (2000). Introduction to the CoNLL-2000 Shared Task: Chunking. In Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning (CoNLL), 127-132.

[5] Explosion AI. (2022). spaCy: Industrial-strength Natural Language Processing in Python. [Online]. Available: https://spacy.io/

[6] Haddi, E., Liu, X., & Shi, Y. (2019). Deep Learning for Natural Language Processing. Springer.

Halaman ini sengaja dibiarkan kosong