

# Sistem Rekomendasi Pembuatan Sampiran Pantun Menggunakan *Tail Similarity*

Winda Monika <sup>#1</sup>, Arbi Haza Nasution <sup>\*2</sup>, Evizariza <sup>#3</sup>

<sup>#1</sup>Jurusan Ilmu Perpustakaan, Fakultas Ilmu Budaya, Universitas Lancang Kuning

<sup>#3</sup>Jurusan Sastra Indonesia, Fakultas Ilmu Budaya, Universitas Lancang Kuning

<sup>1</sup>windamonika@unilak.ac.id

<sup>3</sup>evizariza@unilak.ac.id

<sup>\*2</sup>Prodi Teknik Informatika, Fakultas Teknik, Universitas Islam Riau

<sup>2</sup>arbi@eng.uir.ac.id

**Abstrak**— Pantun adalah puisi lama bentuk tradisi lisan melayu yang termasuk dalam warisan budaya tak benda. Di samping maraknya kegiatan berpantun di masyarakat, ditemukan bahwa pantun modern saat ini banyak terdapat penyimpangan dimana sebagian tidak mencerminkan nilai-nilai leluhur, tidak sesuai jumlah larik dalam bait, jumlah kata dalam larik yang tidak menentu, sampiran tidak mengantarkan sisi pantun, serta tidak memperhatikan susunan rima pantun. Oleh karena itu, pada penelitian ini dibangun sistem rekomendasi pembuatan sampiran pantun menggunakan *tail similarity*. Algoritma *tail similarity* menyederhanakan permasalahan pencocokan kalimat dengan hanya memperhatikan kemiripan akhiran dari kalimat tersebut yang secara berkala menambah bobot untuk sampiran yang sedang dicocokkan dengan isi yang diinputkan dengan pengecekan tiap huruf dari paling kanan (*tail*) sampiran bergerak ke arah kiri sampai tidak ditemukan kecocokan huruf lagi. Dengan input dua baris isi, sistem akan memberikan rekomendasi sampiran yang sesuai dengan skema rima ab-ab dengan melakukan perbandingan urutan antara kata terakhir dari isi A & isi B dengan kata terakhir dari sampiran A & sampiran B dari seluruh pantun di basis data. Algoritma ini mampu memberikan rekomendasi sampiran pantun dengan tepat sesuai dengan kemiripan kata akhir.

**Kata kunci**— *string comparison*, *pantun*, *sequence comparison*, *tail similarity*

## I. PENDAHULUAN

Keberadaan pantun hari ini mendapatkan pengakuan dunia sebagai tradisi lisan dimana pada pertemuan ke 15 *the Intergovernmental Committee for the Safeguarding of the Intangible Cultural Heritage* yang berlangsung di markas besar UNESCO di Paris, Perancis, mengukuhkan pantun sebagai daftar ke 11 warisan budaya tak benda Indonesia oleh UNESCO. Permodelan metadata untuk digitalisasi *Intangible Cultural Heritage* merupakan langkah awal dalam pelestarian warisan bersejarah tak benda [1]–[3].

Pantun merupakan tradisi lisan yang juga berkembang sesuai dengan peradaban masyarakat. Bagi masyarakat melayu, pantun merupakan bentuk karya sastra yang digunakan untuk menyampaikan pesan berupa nilai-nilai kebudayaan dan agama dalam suatu masyarakat atau yang dikenal dengan istilah tunjuk ajar [4]. Di samping itu, pantun juga digunakan sebagai cara seseorang dalam mengekspresikan perasaan dan keadaan di masyarakat sehingga dalam prakteknya sampiran dan isi menggambarkan situasi saat pantun dituturkan seperti pantun jenaka, pantun nasehat, pantun sindiran dan lain sebagainya. Proses pembuatan pantun merefleksikan hasil kreativitas yang berasal dari kreasi intelektual seseorang dalam mengolah kata [5]. Sehingga pelestarian pantun sangat penting menimbang pelestarian nilai-nilai kearifan turut terkandung di dalamnya.

Realitas yang terjadi di masyarakat kini menunjukkan adanya pergeseran nilai-nilai pewarisan pada pantun yang digunakan dimana materi pantun yang disampaikan hanya sebagai formalitas atau pelengkap saja. Pantun hari ini kehilangan fungsi dan hakikat makna sebagai media tunjuk ajar dan pewarisan nilai-nilai luhur budaya bangsa [6]. Selain itu, pantun modern saat ini sebagian cenderung menyimpang dimana tidak sesuai jumlah larik dalam bait, jumlah kata dalam larik yang tidak menentu, sampiran tidak mengantarkan sisi pantun, serta tidak memperhatikan susunan rima pantun [7]. Pantun modern yang diciptakan lebih banyak untuk berkelakar, sebagai pengantar sambutan, maupun kata penutup. Pergeseran kebudayaan ini menjadikan pembuatan pantun sesuai kaidah terbilang sulit bagi penutur. Teknologi yang membantu dalam membuat pantun sangat diperlukan dalam memberikan rekomendasi padanan pantun yang sesuai dengan kaidah tanpa menafikan nilai-nilai leluhur yang semestinya terkandung.

Pantun memiliki aturan atau kaidah pembuatan pantun yaitu terdiri atas sampiran, isi, dan pola a-b-a-b [8], [9]. Sehingga dalam pembuatan sistem rekomendasi

pembuatan sampiran pantun, tiap larik (sampiran dan isi) harus dibandingkan dan terikat pada akhiran pola a-b-a-b.

Pemrosesan bahasa alamiah (*Natural Language Processing*) atau pemrosesan text dan metode *sequence comparison* pada khususnya telah banyak diimplementasikan guna memecahkan masalah di kehidupan sehari-hari seperti perbandingan urutan protein dan DNA [10] dan pencarian cognate pada linguistik sejarah komputasional [11]–[15]. Metode *sequence comparison* merupakan pengurutan karakter yang mencakup tiga model pengurutan yaitu *traces*, *alignment*, dan *listing*. Ketiga model ini mengurutkan kata serta membandingkan kemiripan karakter. Pada model ini diterapkan beberapa algoritma seperti algoritma *global sequence alignment*, algoritma *local sequence alignment*, dan *Levenshtein distance* atau dikenal dengan *edit distance*. Algoritma *global sequence alignment* mengurutkan karakter secara keseluruhan, berbeda dengan algoritma *sequence alignment* yang mengurutkan dan mencari kemiripan karakter bisa parsial (bagian depan, tengah, dan belakang) dimana konsep yang sama juga dilakukan oleh *Levenshtein distance*, dengan perbedaan yaitu *Levenshtein distance* melakukan *remove*, *replace*, dan *insert* untuk karakter yang diurutkan. Algoritma di atas membandingkan kata atau kalimat secara keseluruhan, namun, dalam kasus pembuatan sampiran pantun, hanya akhiran pantun saja yang perlu dibandingkan untuk melihat pola akhiran rima a-b-a-b. Oleh karena itu, diperlukan algoritma baru yang dapat secara khusus menyelesaikan masalah pencocokan rima pantun.

## II. METODOLOGI PENELITIAN

### A. Analisa Karakteristik Pantun

Penelitian ini mengkaji karakteristik pantun yang sesuai kaedah, yaitu dua baris pertama merupakan pembayang atau sampiran, dua baris berikutnya mengandung isi dan terdiri dari empat baris dengan skema rima ab-ab. Dengan input dua baris isi, sistem akan memberikan rekomendasi sampiran yang sesuai dengan skema rima ab-ab.

### B. Pengumpulan Data

Data sekunder berupa naskah pantun didapatkan melalui studi literatur yang diperoleh dari buku-buku pantun cetak dan digital. Naskah pantun cetak kemudian akan melalui proses digitalisasi sehingga dapat dibaca oleh mesin. Data digital naskah pantun kemudian akan disimpan ke dalam basis data MySQL. Terdapat 142 pantun yang diinputkan di database yang selanjutnya digunakan untuk menganalisa keakuratan sistem.

### C. Algoritma Tail Similarity

Algoritma *Tail Similarity* menyederhanakan permasalahan pencocokan kalimat dengan hanya memperhatikan kemiripan akhiran dari kalimat tersebut. Algoritma *Tail Similarity* berupa fungsi rekursif yang

secara berkala menambah bobot untuk sampiran yang sedang dicocokkan dengan isi yang diinputkan dengan pengecekan tiap huruf dari paling kanan (*tail*) sampiran bergerak ke arah kiri sampai tidak ditemukan kecocokan huruf lagi. Detail algoritma *Tail Similarity* dapat dilihat pada Algoritma 1.

Algoritma 1. Pencocokan Bunyi Rima dengan *Tail Similarity*

```
def tailsimilarity(word1, word2, tail1, tail2, bobot):
    if word1 == word2:
        return len(word1)
    if len(word1) == 0 or len(word2) == 0:
        return bobot
    if word1[-1] == word2[-1]:
        tailsimilarity(word1[:-1], word2[:-1], word1[-1],
            word2[-1], bobot+1)
    else:
        if tail1 in ['a','i','u','e','o']:
            return bobot
        else:
            return 0
    return tailsimilarity(word1[:-1], word2[:-1],
        word1[-1], word2[-1], bobot+1)

def matchpantun(sampiran1, sampiran2, isi1, isi2):
    bobotA = 0
    bobotB = 0
    k_sampiran1 = sampiran1.split(' ')[-1]
    k_sampiran2 = sampiran2.split(' ')[-1]
    k_isi1 = isi1.split(' ')[-1]
    k_isi2 = isi2.split(' ')[-1]

    bobotA = tailsimilarity(k_sampiran1, k_isi1, '#', '#', 0)
    bobotB = tailsimilarity(k_sampiran2, k_isi2, '#', '#', 0)

    if bobotA == 0 or bobotB == 0:
        totalBobot = 0
    else:
        totalBobot = bobotA + bobotB

    return totalBobot
```

Agar sistem dapat memberikan rekomendasi sampiran yang sesuai dengan skema rima ab-ab, maka kata terakhir pada sampiran A dibandingkan dengan kata terakhir pada isi A, dan begitu juga kata terakhir pada sampiran B perlu dibandingkan dengan kata terakhir pada isi B.

Kemudian dilakukan perhitungan bobot berdasarkan jumlah karakter yang cocok secara berurutan dengan urutan terbalik. Sebagai contoh pada Gambar 1, bobot perbandingan urutan kata terakhir pada isi A “berju-ang” dan kata terakhir pada sampiran 1 A “lad-ang” memiliki bobot 3, bobot perbandingan urutan kata terakhir pada isi B “t-u-l-a-n-g” dan kata terakhir pada sampiran 1 B “m-a-n-d-i” memiliki bobot 0, sehingga total bobot sampiran

pantun 1 ialah 3. Sedangkan bobot sampiran pantun 2 ialah 6 dengan masing-masing bobot sampiran 2 A ialah 3 dan bobot sampiran 2 B ialah 3.



Gambar. 1 Pencocokan Rima Isi dengan Sampiran Pantun dari Basis Data

D. Perhitungan Penerapan Sistem

Untuk mengukur kemampuan penerapan sistem dengan jumlah pantun di basis data sebanyak 142 pantun, dilakukan pengujian dengan menggunakan kata dasar bahasa Indonesia sebanyak 28.526 kata. Evaluasi sistem rekomendasi sampiran dilakukan dengan menghitung kombinasi kata menggunakan perkalian kartesian (*cartesian product*) dari jumlah akhiran sampiran 1 yang ditemukan cocok dengan akhiran kata dasar menggunakan *algoritma tail similarity*. Perhitungan kombinasi isi pantun digambarkan pada Algoritma 2.

Algoritma 2. Perhitungan kombinasi isi pantun

```

wordList ← input 28.526 kata dasar
totalCombination = 0
notMatched = 0
for s in sampiran:
    counter1 = 0
    counter2 = 0
    for w in wordList:
        if w[tailALength:] == s[0]:
            counter1 += 1
    for w in wordList:
        if w[tailBLength:] == s[1]:
            counter2 += 1
    combination = counter1 * counter2
    totalCombination += combination
if combination == 0:
    notMatched += 1
    
```

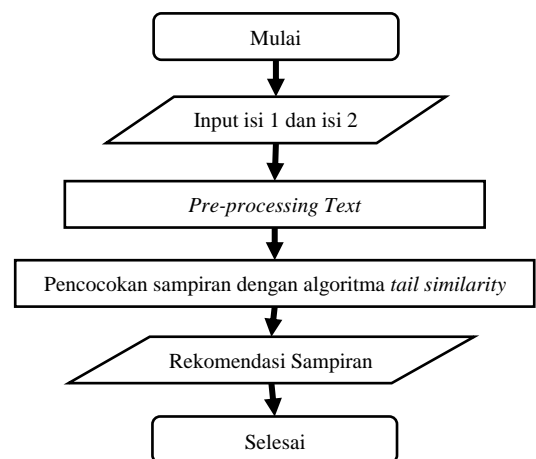
III. HASIL DAN ANALISIS

A. Sistem Rekomendasi Berbasis Web

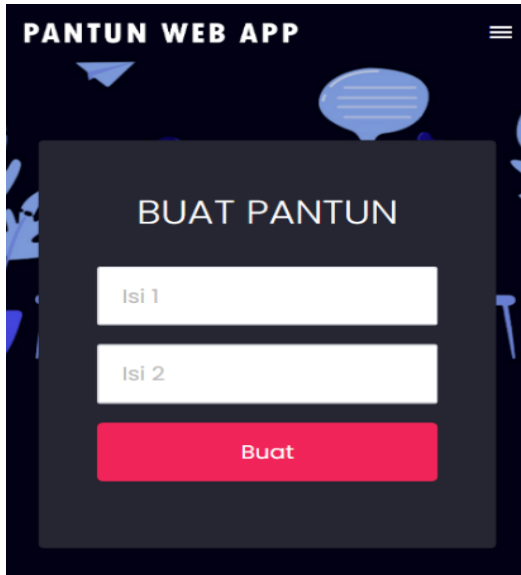
Sistem rekomendasi sampiran berbasis web dibangun dengan dua fitur utama, yaitu: a) fitur “Buat Pantun” yang memudahkan pengguna membuat pantun hanya dengan memberikan input isi 1 dan isi 2 yang kemudian akan dicari sampiran yang sesuai ritma ab-ab dengan menggunakan algoritma *tail similarity*, dan b) fitur “Tambah Pantun” yang memudahkan pengguna untuk berkontribusi menambahkan pantun lengkap dengan sampiran 1, sampiran 2, isi 1 dan isi 2.

Adapun *workflow* atau rangkaian kerja sistem rekomendasi untuk fitur “Buat Pantun” seperti yang terlihat pada Gambar 2 antara lain:

- 1) Input Data/Data Entry  
Pengguna dapat menginputkan isi 1 dan isi 2 seperti ditunjukkan pada Gambar 3.
- 2) Pre-processing Text  
Isi 1 dan isi 2 akan melalui tahap
  - *Casefolding* dimana huruf kapital pada teks tersebut akan diubah bentuknya menjadi huruf kecil,
  - *Remove punctuation* dimana menghapus tanda baca yang tidak sengaja terinputkan
- 3) Pencocokan sampiran  
Dengan algoritma *Tail Similarity*, seluruh pantun yang terdapat di basis data akan dibandingkan kecocokan sampiran 1 dan sampiran 2 dengan isi 1 dan isi 2 yang telah diinputkan.
- 4) Rekomendasi sampiran  
Hasil pencocokan berupa rekomendasi sampiran dari pantun dengan hasil pembobotan kemiripan tertinggi seperti yang ditunjukkan pada Gambar 4.



Gambar. 2 Diagram Kerangka Kerja Sistem Rekomendasi Pembuatan Sampiran Pantun



Gambar. 3 Tampilan Antarmuka Fitur Buat Pantun (Input)



Gambar. 4 Tampilan Antarmuka Fitur Buat Pantun (Output)

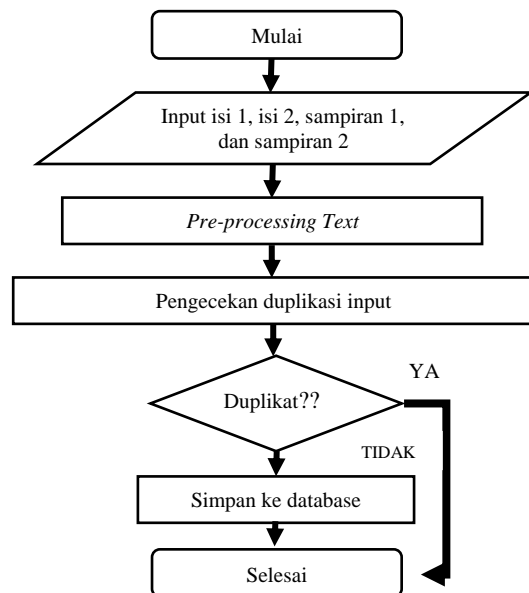
Adapun *workflow* atau rangkaian kerja sistem rekomendasi untuk fitur “Tambah Pantun” seperti yang terlihat pada Gambar 5. antara lain.

- 1) **Input Data/Data Entry**  
Pengguna dapat menginputkan sampiran 1, sampiran2, isi 1, dan isi 2 seperti yang ditunjukkan pada Gambar 6.
- 2) **Pre-processing Text**  
Isi 1 dan isi 2 akan melalui tahap

- *Casefolding* dimana huruf kapital pada teks tersebut akan diubah bentuknya menjadi huruf kecil,
  - *Remove punctuation* dimana menghapus tanda baca yang tidak segera terinputkan
- 3) **Pengecekan duplikasi**  
Data pantun yang diinputkan harus unik entry atau tidak diperbolehkan duplikasi. Apabila ternyata input telah tersedia di database maka penginputan akan gagal.

Adapun *workflow* atau rangkaian kerja sistem rekomendasi untuk fitur “Tambah Pantun” seperti yang terlihat pada Gambar 5. antara lain.

- 2) **Input Data/Data Entry**  
Pengguna dapat menginputkan sampiran 1, sampiran2, isi 1, dan isi 2.
- 2) **Pre-processing Text**  
Isi 1 dan isi 2 akan melalui tahap
  - *Casefolding* dimana huruf kapital pada teks tersebut akan diubah bentuknya menjadi huruf kecil,
  - *Remove punctuation* dimana menghapus tanda baca yang tidak segera terinputkan
- 3) **Pengecekan duplikasi**  
Data pantun yang diinputkan harus unik entry atau tidak diperbolehkan duplikasi. Apabila ternyata input telah tersedia di database maka penginputan akan gagal.



Gambar. 5 Diagram Kerangka Kerja Sistem Rekomendasi Tambahkan Pantun

TABEL I  
PERHITUNGAN KOMBINASI REKOMENDASI ISI PANTUN

#Tail untuk sampiran 1	#Tail untuk sampiran 2	Total Bobot	Kombinasi isi yang dapat direkomendasikan	Jumlah sampiran yang tidak dapat diberikan rekomendasi isi
6	6	12	106	51
6	5	11	318	42
5	6	11	316	43
6	4	10	1.628	33
5	5	10	1.075	33
4	6	10	1.402	34
6	3	9	15.503	31
5	4	9	6.003	19
4	5	9	5.140	22
3	6	9	11.133	32
5	3	8	55.682	16
4	4	8	29.981	7
3	5	8	46.300	20
4	3	7	288.840	3
3	4	7	304.188	5
3	3	6	1.754.319	0



Gambar. 6 Tampilan Antarmuka Fitur Tambah Pantun (Output)

**B. Evaluasi Sistem**

Sistem menerapkan *rule-based* sehingga akan memberikan rekomendasi kata akhir sampiran sesuai dengan pembobotan yang diterapkan oleh algoritma *tail similar*.

Berdasarkan kombinasi kata di basis data yaitu 142 sampiran dan kamus *monolingual* bahasa Indonesia sebanyak 28.526 kata, diperoleh hasil seperti yang terlihat pada Tabel 1. Hasil eksperimen di atas menunjukkan untuk jumlah akhiran kata (*tail*) terbesar yaitu sebanyak 6 huruf akhiran kata sampiran 1 dan sampiran 2, kombinasi isi yang dapat direkomendasikan ialah paling sedikit, yaitu sebanyak 106, dan ditemukan 51 sampiran yang tidak dapat diberikan rekomendasi isi. Namun, untuk jumlah akhiran kata (*tail*) terkecil yaitu sebanyak 3 huruf akhiran kata sampiran 1 dan sampiran 2, kombinasi isi yang dapat direkomendasikan ialah paling banyak, yaitu sebanyak 1.754.319, dan tidak ada ditemukan sampiran yang tidak dapat diberikan rekomendasi isi. Oleh karena itu, dapat ditentukan bahwa jumlah akhiran kata (*tail*) yang paling efisien diimplementasikan ialah sebanyak 3 huruf akhiran kata sampiran 1 dan sampiran 2 sehingga memperluas kemungkinan rekomendasi isi yang dapat diberikan.

#### IV. KESIMPULAN

Sistem rekomendasi pembuatan sampiran dengan algoritma *tail similarity* mampu memberikan rekomendasi pembuatan sampiran pantun dengan tepat sesuai dengan kemiripan kata akhir. Dengan sistem pembobotan, akhir kata sampiran 1 – isi 1 dan sampiran 2 – isi 2, bobot tertinggi berkemungkinan besar untuk dipilih atau direkomendasikan oleh sistem karena kemiripan katanya. Namun dikarenakan keterbatasan jumlah pantun di basis data, berdasarkan evaluasi dengan mengkombinasikan dengan kamus *monolingual* bahasa Indonesia, *recall* kata akhir untuk bobot tertinggi masih rendah, oleh karena itu, diperlukan lebih banyak lagi dokumentasi pantun. Dari hasil eksperimen ditemukan bahwa jumlah akhiran kata (*tail*) yang paling efisien diimplementasikan ialah sebanyak 3 huruf akhiran kata sampiran 1 dan sampiran 2 sehingga memperluas kemungkinan rekomendasi isi yang dapat diberikan. Sistem rekomendasi pembuatan sampiran pantun membantu pengguna dalam membuat pantun dengan jumlah larik dalam bait yang sesuai kaidah pantun dan sesuai susunan rima pantun dengan efektif, efisien, dan tetap mempertahankan kaidah-kaidah pantun yang berlaku.

#### UCAPAN TERIMA KASIH / ACKNOWLEDGMENT

Penelitian ini didanai oleh skim APBU Tahun Anggaran 2020/2021 Lembaga Penelitian dan Pengabdian Pada Masyarakat Universitas Lancang Kuning.

#### REFERENSI

- [1] C. Wijesundara, W. Monika, and S. Sugimoto, "A metadata model to organize cultural heritage resources in heterogeneous information environments," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Nov. 2017, vol. 10647 LNCS, pp. 81–94, doi: 10.1007/978-3-319-70232-2\_7.
- [2] W. Monika, C. Wijesundara, and S. Sugimoto, "Modeling Digital Archives of Intangible Cultural Heritage based on One-to-One Principle of Metadata," in *Proceedings of the 8th Asia-Pacific Conference on Library & Information Education and Practice (A-LIEP 2017)*, 2017.
- [3] S. Sugimoto, S. Kiryakos, C. Wijesundara, W. Monika, T. Mihara, and M. Nagamori, "Metadata Models for Organizing Digital Archives on the Web: Metadata-Centric Projects at Tsukuba and Lessons Learned," in *International Conference on Dublin Core and Metadata Applications*, Nov. 2018, vol. 0, pp. 95–105.
- [4] T. Tarwiyani, M. Munir, and S. B. Trisakti, "Pantun as a Means of Character Education in the Life of the Nation and State," in *3rd International Conference on Learning Innovation and Quality Education (ICLIQE 2019)*, 2020, pp. 1321–1332.
- [5] T. D. Leoni, "Pantun sebagai Akar Literasi dan Pembentukan Moral Anak di Era Disrupsi," in *Prosiding Seminar Nasional Pascasarjana (PROSNAMPAS)*, 2019, vol. 2, no. 1, pp. 867–870.
- [6] T. Effendy, *Tunjuk ajar dalam pantun Melayu*. Balai Kajian dan Pengembangan Budaya Melayu, 2004.
- [7] R. Setyadiharja, *Khazanah Negeri Pantun*. Deepublish, 2020.
- [8] T. Andriani, "Pantun dalam Kehidupan Melayu (pendekatan historis dan antropologis)," *Sos. Budaya*, vol. 9, no. 2, pp. 195–211, 2012.
- [9] Z. H. Mubarak, "Struktur dan Fungsi Pantun Pembuka dan Penutup Majelis dalam Adat Melayu," in *Prosiding Seminar Nasional Ilmu Sosial dan Teknologi (SNISTEK)*, 2021, vol. 3, pp. 168–173.
- [10] W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison.," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 85, no. 8, pp. 2444–2448, Apr. 1988, doi: 10.1073/pnas.85.8.2444.
- [11] J.-M. List, *Sequence comparison in historical linguistics*, vol. 1. düsseldorf university press, 2014.
- [12] J.-M. List, M. Walworth, S. J. Greenhill, T. Tresoldi, and R. Forkel, "Sequence comparison in computational historical linguistics," *J. Lang. Evol.*, vol. 3, no. 2, pp. 130–144, Jul. 2018, doi: 10.1093/jole/lzy006.
- [13] A. H. Nasution, Y. Murakami, and T. Ishida, "A Generalized Constraint Approach to Bilingual Dictionary Induction for Low-Resource Language Families," *Acm Trans. Asian Low-Resource Lang. Inf. Process.*, 2018, doi: 10.1145/3138815.
- [14] A. H. Nasution, Y. Murakami, and T. Ishida, "Plan Optimization to Bilingual Dictionary Induction for Low-resource Language Families," *Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 20, no. 2, Mar. 2021, doi: 10.1145/3448215.
- [15] A. H. Nasution, Y. Murakami, and T. Ishida, "Constraint-based bilingual lexicon induction for closely related languages," in *Proceedings of the 10th International Conference on Language Resources and Evaluation, LREC 2016*, 2016, [Online]. Available: <http://www.scopus.com/inward/record.url?eid=2-s2.0-85034636317&partnerID=MN8TOARS>.