

Article

Determining Intermediary Closely Related Languages to Find a Mediator for Intertribal Conflict Resolution

Arbi Haza Nasution ^{1,*} , Shella Eldwina Fitri ¹, Rizauddin Saian ² , Winda Monika ³ 
and Nasreen Badruddin ⁴ 

¹ Department of Informatics Engineering, Universitas Islam Riau, Pekanbaru 28284, Riau, Indonesia

² College of Computing, Informatics and Media, Universiti Teknologi MARA (UiTM) Perlis Branch, Arau 02600, Perlis, Malaysia

³ Department of Library Information, Universitas Lancang Kuning, Pekanbaru 28266, Riau, Indonesia

⁴ Department of Electrical and Electronic Engineering, Universiti Teknologi PETRONAS, Seri Iskandar 32610, Perak, Malaysia

* Correspondence: arbi@eng.uir.ac.id

Abstract: Indonesia has a diverse ethnic and cultural background. However, this diversity sometimes creates social problems, such as intertribal conflict. Because of the large differences among tribal languages, it is often difficult for conflicting parties to dialog for conflict resolution. To address this problem, we aim to find intermediary closely related languages from a language similarity knowledge graph using the best-performing pathfinding algorithms. In this research, we analyze the performances of two pathfinding algorithms, namely, Dijkstra and Yen's K, by comparing their execution time and the total lexical distances of the intermediary languages (called "the cost"). Our research findings show that even though the Dijkstra and Yen's K algorithms have equal total cost for all the cases, Yen's K outperformed Dijkstra at searching for intermediary languages that are closely related, with an average of 160% higher performance on execution time. The selection of native speakers of the obtained intermediary languages as mediators is formalized as an optimization problem with four criteria: language similarity, geographical distance, background, and expected salary. We present a case study where the intermediary closely related languages can be used as a guideline to find mediators who can help resolve the intertribal conflicts among Indonesian tribes. To calculate the first criteria, we implemented the Yen's K algorithm to calculate the shortest path between target languages and return the path via the intermediary languages. This implementation shows the potential use of the mediator selection model defined in this paper in various other roles such as trader or salesman, politician's spokesman, reporter or journalist, etc.

Keywords: closely related languages; intertribal conflict; conflict resolution; pathfinding algorithms



Citation: Nasution, A.H.; Fitri, S.E.; Saian, R.; Monika, W.; Badruddin, N. Determining Intermediary Closely Related Languages to Find a Mediator for Intertribal Conflict Resolution. *Information* **2022**, *13*, 557. <https://doi.org/10.3390/info13120557>

Academic Editor: Costas Vassilakis

Received: 18 September 2022

Accepted: 22 November 2022

Published: 28 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The national motto of Indonesia is *Bhinneka Tunggal Ika*, which means "unity in diversity", which clearly underlines the fact that Indonesia is a country with a diverse ethnic and cultural background. However, this diversity sometimes creates social problems, such as intertribal conflicts. Panggabean, Indonesia's prominent conflict resolution expert [1], states that the number of intertribal conflicts which occurred in Indonesia between 1990 and 2003 is 2608, with the death toll approximately around 10.758% or 96.4% from the total population of the conflicted areas. Such conflicts have taken place in several areas, for example, there were conflicts between the Dayak and Madura tribes, Sambas riots in 1999 [2], conflict between the Christian Ambonese-ethnic and Javanese and Makassar-ethnic migrants, mostly Muslims in 2002, Balinese and Lampung ethnics conflict in 2012 [3], and many more. Most of them are caused by structural conflict, interest conflict, relationship, social-psychology and prejudices conflict, local and traditional values conflict, data conflict [4], political and economic disadvantages [5], and intercultural interaction and communication problems [6].

Intertribal conflict usually is started from personal conflicts that escalate into local, national, and even global ones. Some studies suggest several methods to overcome these conflicts, such as by finding mediators to reconcile the conflicting parties to reach an agreement [5,7], by face-to-face negotiating based on local wisdom or culture [3], by enhancing intercultural competence [3], by comparative analysis of the language [8], and so on.

Language barriers are assumed as a major obstacle in communication. If people from tribe A can communicate using only the A language, and people from tribe B can communicate using only the B language, then communication cannot occur without a neutral mediator who understands both languages. The mediator can belong to tribe C that communicates using the C language, which is closely related to both the A language and B language. To choose a mediator, the language of the mediator needs to be similar to the languages of the conflicting parties.

To address this problem, we aim to find intermediary closely related languages from a language similarity knowledge graph using the best-performing pathfinding algorithms. To show the importance of the intermediary closely related languages, we present a case study where we use the intermediary closely related languages as a guideline to find mediators who can help resolve the intertribal conflicts among Indonesian tribes.

In the following sections, we review the relevant literature on tribal conflicts and pathfinding. Then, we present our data collection methods and results. In this study, the tribal languages used in the experiment are not actually that of a tribe in conflict. They are chosen for the sake of variety of the simulation. Finally, we discuss our conclusions and provide recommendations for further studies on this topic.

2. Literature Review

2.1. Tribe Diversity and Intertribal Conflict

Indonesia has a population consisting of persons from different nationalities, religions, ethnicities, and languages. According to the 2010 statistical data, 1340 ethnic groups are spread throughout Indonesia.

Tribal groups are ethnic groups and community cultures that are formed from generation to generation as part of the community's cultural system. The tribe identity and attributes of a community group will be inherited by the next generation. Culturally, tribe identity and attributes are directly attached to each person according to the parents' tribes [9]. According to Mulyana [10], tribes in Indonesia are usually located in various regions, for example, the Sundanese are in West Java, the Javanese in Central and East Java, the Batakese in North Sumatra, the Ambonese in Maluku, and the Buginese in South Sumatra.

Conflicts are unavoidable in societies and organizations. Mismatches in social processes can cause conflicts. Theoretically, conflict is defined as a condition in which a dispute occurs between one party and one or more parties that have different views or interests. Conflict is also a form of struggle to obtain intangible resources such as value, status, power, and authority. In such cases, the conflicting parties are not only in conflict to gain benefits for themselves, but they also aim to subdue their rivals [11].

Conflicts are an inherent part of human life and are often unavoidable. When humans are faced with life choices, they might have to act contrary to their conscience (intrapersonal) and/or act against other humans (interpersonal), which leads to conflict. Conflicts become serious when individuals hold strong negative views that render them incapable of conflict management and lead them to violent behaviors [12].

In the Law of the Republic of Indonesia Number 7 of 2012 concerning social conflict handling, it is stated that social conflict is a physical clash and/or clash in which violence occurs between two or more groups of people. This law also specifies that the violence that occurs within a certain period and has a wide impact resulting in insecurity and social disintegration would be considered a social conflict. A social conflict would also disrupt national stability and hinder social development. Conflicts can occur at any time and can involve anyone; they can also occur for any cause. A person can even become involved in a conflict that is happening around them because of some misunderstanding or differences of opinions, customs, cultures, traditions, and ethnicities.

Indonesia is a culturally diverse country with approximately 1340 tribes. Sometimes, tribe diversities can trigger social problems, such as tribal conflicts. The main causes of tribal conflicts are social inequalities, economic problems, and political differences. According to Mulyana [10], the occurrence of tribal conflicts is closely related to the historical writings of unification and uniformity of mono-cultural nationalism. The Indonesian government has enforced centralization that has resulted in the loss of local identities. A mono-cultural and centralized thinking has been indoctrinated into citizens. However, the formation of a nation should start from local ethnic dynamics. Local events that occur must be positioned as events that are autonomous and unique, which become the basis for the formation of a nation. The values of nationalism have been questioned when several ethnic conflicts emerged, such as those in Sampit, Maluku, and Poso, in addition to the ethnic resistance to the central power.

Every domestic conflict resolution does not necessarily depend on the national law enforcement institutions and apparatuses, but it is necessary to have open spaces and involve the local community in the conflict resolution process. However, the implementation of conflict resolution is not always easy, especially when cross-cultural communication is involved, because the parties with different cultural backgrounds must have the same frame of reference to effectively respond to a problem. Therefore, cross-cultural communication is very important in conflict resolution [13].

In the Law of the Republic of Indonesia Number 7 of 2012 concerning social conflict handling, it is stated that conflict resolution is a series of activities performed in a systematic and planned manner in situations and events both before, during, and after a conflict. These activities include conflict prevention, conflict cessation, and post-conflict recovery. Conflict prevention is a series of activities performed to prevent conflicts by increasing the institutional capacity and early warning systems. Conflict cessation is a series of activities to end violence, save victims, limit the expansion and escalation of the conflict, and decrease the number of victims and property losses.

Conflict management requires skills, such as effective communication, problem solving, and functional skills that can increase productivity. Conflict resolution is not easy. Whether a conflict is resolved quickly or not depends on the willingness and openness of the disputing parties to resolve the conflict, the severity level of the conflict, and the ability of third parties (who are involved in conflict resolution) to intervene. Since one of the potential cause of intertribal conflict is intercultural interaction and communication problems [6], we need to make good use of the language diversity to overcome the problem.

2.2. Closely Related Languages

Language is a system of arbitrary sound symbols used by a community to cooperate, interact, and identify themselves. Therefore, language here is a means of communication in social life; it is both written and oral. Without language, humans cannot interact with other humans. Closely related languages have the same origin or protolanguage and usually belong to the same language family. According to Gooskens et al. [14], linguistic diversity can lead to communication problems that might be overcome only with sufficient knowledge about the language situation at hand. The principle of receptive multilingualism is based on the fact that certain language pairs are so closely related that the speakers can communicate with each other using their own language and without any prior language instructions. This strategy is widely used for communication among speakers of the three mainland Scandinavian languages: Danish, Swedish, and Norwegian. For example, Danish tourists traveling to Sweden will often speak in their mother tongue, Danish, to the Swedes that they meet en route [15]. The Swedes often respond hesitantly at first in Danish, but they soon discover that it is possible and even easier to respond in their own mother tongue, Swedish, than in Danish.

Comparative linguistics is a branch of historical linguistics that is concerned with language comparisons to determine the historical relatedness and construct language families [16]. The genetic relationship of languages is used to classify languages into language families. Lexicostatistical comparisons explain the historical relationships between languages by estimating the percentage of related words in language pairs. For example, Germanic languages are more

closely related to one another than to Romance languages, and vice versa. In the lexicostatistical approach, the percentage of cognates shared by two languages is estimated based on cognacy judgments made by experts [17].

The vocabulary used for such cognacy judgments often consists of translation pairs from Swadesh lists. A Swadesh list is a classic compilation of basic concepts for historical and comparative linguistics. Swadesh lists are small sets of universal culture-free meanings that are robust to changes in meanings and appearances over time. The meanings of items in Swadesh lists are considered resistant to borrowings or chance resemblances among languages. Quantifications of the percentage of shared cognates in Swadesh lists can accurately predict language relatedness [18].

Therefore, we concluded that a language can be considered closely related to a target language if it has similarities with the target language [19–23]. In this study, a language is considered closely related to another language if it has a high similarity value with it. This relationship is useful for finding mediators in resolving conflicts between tribes that speak different languages.

2.3. Automated Similarity Judgment Program (ASJP)

On the ASJP official website, it is stated that ASJP aims to include 40 word lists from all languages of the world. Obtaining lexical distance by comparing lists of words is useful, for example, for classifying a language group and for inferring the ages of differences.

The Automated Similarity Judgment Program (henceforth ASJP) is a project dedicated to the diachronic analysis of the world's linguistic diversity, including the specific task of language classification. A set of 40 highly stable lexical items was selected and, subsequently, a large database of word lists with translational equivalents of these 40 items (or, minimally 70% of the items) in the majority of the world's languages was assembled [24]. The word lists are transcribed in a simplified ASCII representation already described in several papers [25–27]. Since 2008, the preferred approach to computing distances among languages for further input to various analyses has been a modified version of the Levenshtein or 'edit' distance called LDND [25,28].

In research conducted by Müller et al. [26], graphically, the world language tree illustrates relative degrees of lexical similarity holding among 4350 of the world's languages and dialects (henceforth, languages) currently found in the ASJP database. Four factors influence lexical similarity registered in the tree: (1) genetic or genealogical relationship of languages, (2) diffusion (language borrowing), (3) universal tendencies for lexical similarity such as onomatopoeia, and (4) random variation (chance). Languages branched closely together on the tree may be so because of strong lexical similarity produced by any one or a combination of the four factors.

Calculating the Levenshtein distance between translated words from the Swadesh list, then taking the average value from the calculation is a way to obtain the similarity value between languages. Levenshtein distance (LD) is a measure of the similarity between two strings measured from the number of deletions, insertions, or substitutions required. The Levenshtein distance algorithm is shown in Table 1.

Table 1. Levenshtein distance algorithm.

Step	Description
1	Set n to be the length of s . Set m to be the length of t . If $n = 0$, return m and exit. If $m = 0$, return n and exit. Construct a matrix containing $0..m$ rows and $0..n$ columns.
2	Initialize the first row to $0..n$. Initialize the first column to $0..m$.
3	Examine each character of s (i from 1 to n).
4	Examine each character of t (j from 1 to m).
5	If $s[i]$ equals $t[j]$, the cost is 0. If $s[i]$ does not equal $t[j]$, the cost is 1. Set cell $d[i,j]$ of the matrix equal to the minimum of:
6	<ul style="list-style-type: none"> • The cell immediately above plus 1: $d[i-1,j] + 1$. • The cell immediately to the left plus 1: $d[i,j-1] + 1$. • The cell diagonally above and to the left plus the cost: $d[i-1,j-1] + \text{cost}$.
7	After the iteration steps (3, 4, 5, 6) are complete, the distance is found in cell $d[n,m]$.

2.4. Pathfinding Algorithms

The pathfinding algorithm is built on the graph search algorithm by tracing the route from one node to another node, that is, traversing the route associated with other nodes until it reaches the destination node. A pathfinding algorithm is used to identify the optimal routes that can be used for logistics planning, call routing, or low-cost IP, including game simulations [29].

Pathfinding is a process that determines how to travel from a source to a destination in a graph [30]. A graph consists of several arcs connecting certain nodes. A graph with labels can have more than one description attached to each node, which differentiates among the graph nodes. Dijkstra is the most common pathfinding algorithm in the computer science literature. Dijkstra is applied on a weighted graph to find the shortest path in the graph using the total weight between each pair of nodes. Several other algorithms have been developed for problem variants, including the directed and undirected edges. The graph search is divided into blind search and heuristic search [31]. In this study, we used the Dijkstra and Yen's K pathfinding algorithms to calculate the shortest path between a pair of nodes to find the intermediary closely related languages.

2.4.1. Dijkstra Algorithm

The Dijkstra algorithm calculates the shortest (weighted) path between a pair of nodes. In this category, Dijkstra's algorithm is the most well-known. It is a real-time graph algorithm and can be used as part of the normal user flow in a web or mobile application.

Dijkstra's algorithm visits vertices in the graph one by one, starting with the object's starting point. It then examines the closest vertex which is yet to be examined, and this process runs in an outer loop which terminates when either the vertex examined happens to be the target or else if the target is not found, even after all the vertices have been examined. Otherwise, the closest vertices to the examined vertex are then added to the collection of vertices to be examined. In this fashion, it expands outwards from the starting point until it reaches the goal. When the target is found, the loop terminates, and then the algorithm backtraces its way to the start, remembering the required path. Finding the Dijkstra starting from the starting point to the destination point is how the Dijkstra algorithm works. However, this algorithm is not recommended to find a target or target, because this algorithm must examine a number of nodes, which results in spending extra time and resources because the number of nodes to be checked will continue to increase. However, if there already is a target or destination to look for, this algorithm will serve as the quickest option in finding the shortest path [32].

Dijkstra, which is useful for finding the optimal route between a node and the destination node, is widely used to find the shortest path between locations, for example, finding the shortest path from a company to the hospital. In this case, finding the shortest pathway is useful for efficient travel time, so that the time needed to get to the hospital is less. Example use cases include the following [29]:

- Finding directions between locations. The Dijkstra algorithm is applied to Google Maps to provide directions and find the shortest path that connects the starting location to the intended location.
- Finding the degrees of separation between people in social networks. For example, when viewing someone's profile on LinkedIn, it will indicate how many people separate someone in the graph, as well as listing mutual connections. As another example, on Facebook, where when visiting a friend's profile on Facebook we can see other people's Facebook accounts that are suggested, where the account is a friend of our friend on Facebook. Facebook will find the possibility for us to also know that person; this is called friends of friends.
- Finding the number of degrees of separation between an actor and Kevin Bacon based on the movies they have appeared in (the Bacon Number). Bacon Number is a Google feature that shows the actor or actress relationship with Kevin Bacon, with the assumption that every actress or actor has been linked to Kevin through other actors or actresses.

2.4.2. Yen's K Algorithm

The Yen's K-Shortest Paths algorithm is similar to the Dijkstra algorithm, however, the difference is that the algorithm does not only find the shortest path between pairs of nodes. This algorithm can calculate the shortest path up to as many as K paths. This algorithm was invented by Jin Y. Yen in 1971, which he described as "Finding the K Shortest Loopless Paths in a Network". The utility of this algorithm is to obtain the second, third, and so on shortest paths as much as K, which is useful as an alternative path when the first shortest path is not the only desired destination. It is very helpful when more than one backup plan is needed [29].

3. Materials and Methods

3.1. Data Preparation

This study uses a dataset from the research conducted by Nasution and Murakami [33]. In this research, they performed visualization of language similarity clusters by using ASJP to generate language similarities. The dataset consists of 119 Indonesian tribal languages, as shown in Table 2; each language is represented by a node labeled language. Each language node has 16 properties. The link between the nodes is called a relation; a relation has two properties: similarity and distance. Similarity refers to the lexical similitude between any two languages, and the distance is equal to 100 min similarity values as shown in Equation (1). In this study, only three properties were considered to be important, namely distance, name, and coordinates.

$$distance = 100 - similarity \quad (1)$$

Distance is the first important property. This property exists in the relationship between the nodes. To find the shortest path between a pair of nodes, the shortest distance is selected. Languages that are close have large similarities. However, in the pathfinding algorithm, the algorithm will calculate the shortest distance between a pair of nodes as the smallest distance. Consequently, the distance property is used to measure the cost of finding a similar intermediate language.

Table 2. Dataset consisting of 119 Indonesian tribal languages.

No.	Language	No.	Language	No.	Language
1	Abung Sukanda Lampung Nyo	41	Komering	81	Pitulua Bajau
2	Aceh	42	Konjo	82	Pubian Lampung Api
3	Adumanis Ulu Komering	43	Kota Agung Lampung Api	83	Ramau Lampung Api
4	Ambonese Malay	44	Krui Lampung Api	84	Rejang
5	Anaiwoi Bajau	45	Lakaramba Bajau	85	Sadam
6	Bajoe Bajau	46	Lakoena Bajau	86	Salako Badamea
7	Bali	47	Lamaholot Ile Mandiri	87	Samihim
8	Banggai	48	Lampung	88	Sangir
9	Banjarese Malay	49	Lampung Nyo Ambung Kotabumi	89	Sasak
10	Baree	50	Lampung Nyo Melinting	90	Savu
11	Basemah	51	Langgara Laut Bajau	91	Selayar
12	Batak Angkola	52	Lapulu Bajau	92	Sika
13	Batak Mandailing	53	Lauru Bajau	93	Sindue Tawaili
14	Belalau Lampung Api	54	Lemo Bajau	94	Soppeng Buginese
15	Betawi	55	Lewa Kambera	95	Southern Kampera
16	Bima	56	Lio	96	Sukau Lampung Api
17	Boepinang Bajau	57	Lom	97	Sumbawa
18	Buginese	58	Luwuk Bajau	98	Sundanese
19	Coastal Konjo	59	Madurese	99	Sungkai Lampung Api
20	Daya Lampung Api	60	Makasar	100	Tae
21	Delang	61	Malang	101	Talang Padang Lampung Api
22	Ende	62	Malay	102	Tamuan
23	Gayo	63	Mambae	103	Tara
24	Gorontalo	64	Mandar	104	Tetun
25	Iilir Komering	65	Manggarai	105	Toba Batak

Table 2. Cont.

No.	Language	No.	Language	No.	Language
26	Indonesian	66	Menggala Tulang Bawang Lampung	106	Tolaki
27	Indonesian Bajau	67	Minangkabau	107	Tolaki Asera
28	Jabung Lampung Api	68	Mongondow	108	Tolaki Konawe
29	Jambi Malay	69	Moramo ajau	109	Tolaki Laiwui
30	Kadatua	70	Muna	110	Tolaki Mengkongga
31	Kaleroang Bajau	71	Ngaju Baamang	111	Tolaki Wiwirano
32	Kalianda Lampung Api	72	Ngaju Oloh Mangtangai	112	Tontemboan
33	Kambera	73	Ngaju Oloh Mangtangani	113	Tukang Besi Northern
34	Kapuas Kahayan	74	Ngaju Pulopetak	114	Tukang Besi Sothern
35	Karo Batak	75	Nias Northern	115	Uab Meto
36	Katingan	76	Ogan	116	Umbu Ratu Nggai Kambera
37	Kayu Agung Asli Komering	77	Old Or Middle Javanese	117	Way Kanan Lampung Api
38	Kayuadi Bajau	78	Padei Laut Bajau	118	Way Lima Lampung Api
39	Kerinci	79	Palembang Malay	119	Yogyakarta
40	Kolo Bawah Bajau	80	Perjaya Ulu Komering		

3.2. Experiment Design

The pathfinding algorithms that can be used to determine the shortest path between a pair of nodes are the Dijkstra and Yen’s K shortest path algorithms. These algorithms can be used to find the closely related intertribal languages in Indonesia, which will help us find a mediator to resolve tribal conflicts. However, only the algorithm that has the best performance will be selected.

One way of obtaining the similarity value between languages is by calculating the Levenshtein distance (LD) between the translated words from the Swadesh list and then taking the average value of the calculated results. LD is a measure of the similarity between two strings measured from the number of deletions, insertions, or substitutions. In this study, we define the similarity value in the form of a relation property that can be calculated in the algorithm. Similarity property defines the similarities between nodes or between languages. The greater the similarity value, the higher the level of lexical similarity of the language. Conversely, the smaller the similarity value, the lower the level of lexical similarity of the language.

Figure 1 shows an example of the formalization of a graph in the research by Nasution and Murakami [33]. Here, a node represents a language, and an edge represents a language lexical similarity between the two languages. The thickness of an edge represents the similarity between the two languages. For example, in Figure 1, LA can be connected to LZ using two paths: LA–LB–LZ and LA–LC–LZ. Node LA and node LB have a similarity of 40, which means the lexical similarity level value is 40. Node LA and node LC have a similarity of 30, which means the lexical similarity level value is 30. The same holds for the similarity of node LB and node LZ, which is 10, and the similarity of node LC and node LZ, which is 40. Therefore, the total similarity of the path LA–LB–LZ is 50, and the total similarity of the path LA–LC–LZ is 70.

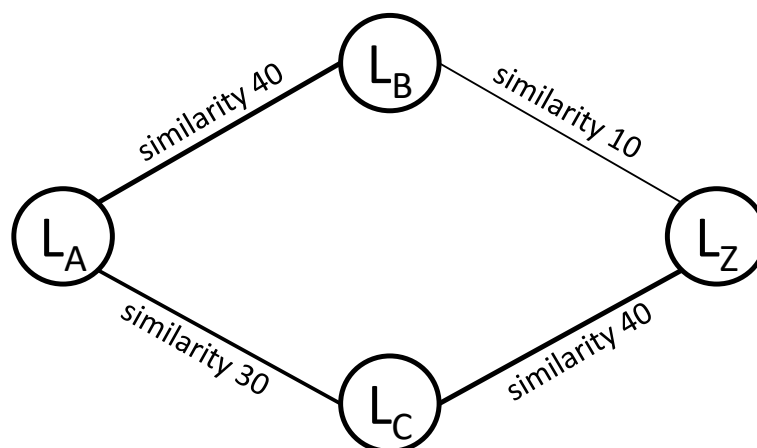


Figure 1. Example of language similarity graph.

The pathfinding algorithm works by selecting the path with the shortest cumulative distance from the node LA to the node LZ. In fact, we wanted to find an intermediate language that was as similar as possible to the source language and target language, which means that the intermediate language needs to have paths with the highest cumulative similarity. Therefore, in this study, we created a property called distance, as shown in Equation (1), hereinafter called “cost”.

Cypher projection was used in this research for the Dijkstra and Yen’s K shortest path algorithms. In this study, the tribal languages used in the experiment were selected randomly for the sake of simulation, not necessarily for belonging to a conflicting tribe.

Listing 1 shows the Cypher projection for the Dijkstra algorithm finding intermediary languages between BALI and PALLU, with a threshold of maximum distance equal to 62. This algorithm declares a start node and an end node representing the source language and the target language, respectively. The algorithm works by tracing the path connecting the two nodes. The algorithm returns the path with the minimum cost.

Listing 2 shows the Cypher projection for the Yen’s K shortest path algorithm finding intermediary languages between BALI and PALLU, with a threshold of maximum distance equal to 62. Similar to the Dijkstra algorithm, at the start of the Yen’s K algorithm, the start node and end node are declared to represent the source language and the target language, respectively. The algorithm works by tracing the path connecting the two nodes. The algorithm returns the path with the minimum cost.

Listing 1. Cypher projection of the Dijkstra algorithm.

```

1 MATCH (start:Language {name: "BALI"}),
2 (end:Language {name: "PALLU"})
3 CALL gds.alpha.shortestPath.stream({
4   nodeQuery:"MATCH(n:Language) RETURN id(n) AS id",
5   relationshipQuery:"MATCH(n:Language)-[r:SIMILAR]-(m:Language)
6   WHERE r.distance < 62
7   RETURN id(n) AS source, id(m) AS target, r.distance AS cost",
8   startNode: start, endNode: end,
9   relationshipWeightProperty:"cost"})
10 YIELD nodeId, cost
11 RETURN gds.util.asNode(nodeId).name AS Language, cost as Cost;

```

Unlike the Dijkstra algorithm, the Yen’s K algorithm has a variable K, and the K value determines the number of shortest paths that can connect the two nodes. The K value is used as a solution to find alternative connected paths, and this value can be adjusted depending on the alternative paths to be obtained. However, in this experiment, only the best path is needed; therefore, the K value was set to 1.

Listing 2. Cypher projection of the Yen’s K shortest path algorithm.

```

1 MATCH (start:Language{name:"BALI"}), (end:Language{name:"PALLU"})
2 CALL gds.alpha.kShortestPaths.stream({
3   nodeQuery:"MATCH(n:Language) RETURN id(n) as id",
4   relationshipQuery:"MATCH (n:Language)-[r:SIMILAR]->(m:Language)
5   WHERE r.distance<62 RETURN id(n) as source, id(m) as target,
6   r.distance as cost",
7   startNode: start,
8   endNode: end,
9   relationshipWeightProperty:"cost",
10  k: 1
11 })
12 YIELD index, sourceNodeId, targetNodeId, nodeIds, costs, path
13 RETURN index, [node in gds.util.asNodes(nodeIds[1..-1]) | node.name] AS
   via,
14 reduce(acc=0.0, cost in costs | acc + cost) AS totalCost;

```

In both Listings 1 and 2 queries, we set the distance threshold as <62, which means the distance of both languages is less than 62 and similarity between languages is bigger than 38.

The pathfinding algorithms for Dijkstra and Yen’s K will return the smallest distance property value that shows the magnitude of the lexical similarity of the two languages.

The next step is to compare the algorithms and find the algorithm the most suitable for finding closely related languages. The algorithms are compared based on their performances on two parameters, i.e., execution time and total cost.

3.3. Mediator Selection as Optimization Problem

After finding the intermediary closely related languages using the pathfinding algorithms, a mediator who speaks those languages can be selected. As shown in Figure 2, mediators can have many roles, including arbitrator of intertribal conflict (as the main case study in this paper), trader or salesman, politician’s spokesman, reporter or journalist, and many other potential roles. Mediators that belong to any one of these roles have the privilege and advantage to do their job due to their ability to understand the target languages better than random people. The selection of native speakers of the obtained intermediary languages as mediators is formalized as an optimization problem with the following criteria:

- C_1 : Average language similarity between the mediator candidate’s native language and the target languages.
- C_2 : Average geographical distance between the mediator candidate’s location and the target languages’ locations.
- C_3 : The mediator candidate’s experience or background to support the mediator role.
- C_4 : The mediator candidate’s expected salary.

For example, as shown in Figure 3, to determine if the mediator between the target languages L_Z and L_C should be selected from L_A or L_B , we need to calculate C_1 by averaging language similarity between L_A , L_Z , and L_C and comparing it with the average of language similarity between L_B , L_Z , and L_C . The same goes for calculating C_2 . Finally, information from C_3 and C_4 can be integrated to calculate the overall cost using a weighted sum model. The weight of each criteria can be defined by an expert for each mediator role.

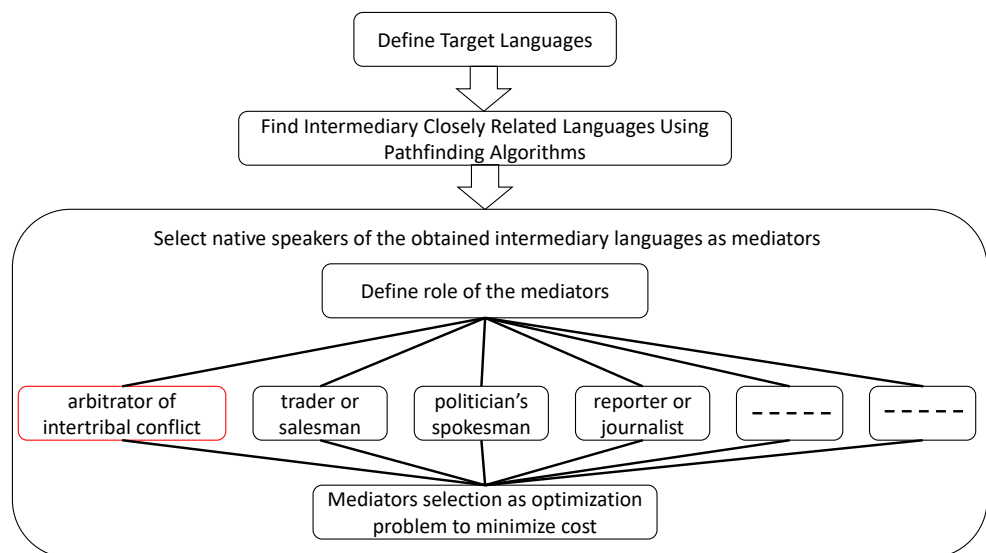


Figure 2. Mediator Selection Model.

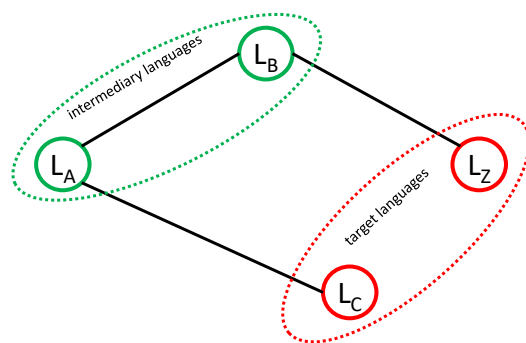


Figure 3. Example of mediator selection from the intermediary languages.

4. Results

4.1. Determining Intermediary Closely Related Languages

4.1.1. Result of Dijkstra Algorithm

Table 3 shows the results of the Neo4j Cypher projection from Bali to Buginese, Ambonese Malay to Karo Batak, and Yogyakarta to Mandar using the Dijkstra algorithm.

Table 3. Cumulative Cost using Dijkstra.

Maximum Distance	Language Pair	Intermediary Languages	Cumulative Cost
61	BALI -BUGINESE	BALI	0.0
		PALEMBANG MALAY	60.9
		EMBALOH	114.37
		BUGINESE	165.77
60	AMBONESE MALAY -KARO BATAK	AMBONESE MALAY	0.0
		TERNATE PASAR	14.77
		KARO BATAK	72.94
		YOGYAKARTA	0.0
63	YOGYAKARTA -MANDAR	PALEMBANG MALAY	62.03
		MAMUJU	123.30
		MANDAR	147.69

The results from Bali to Buginese based on execution time and total cost using the Dijkstra algorithm are 617 ms and 165.77, respectively. The complete route is from Bali to Palembang, Malay to Embaloh, and then to Buginese. The results obtained for the path from Ambonese Malay to Karo Batak based on the execution time and total cost are 730 ms and 147.69, respectively. The route is from Ambonese Malay to Ternate Pasar and then to Karo Batak. The results obtained for the path from Yogyakarta to Mandar based on execution time and total cost are 730 ms and 147.69, respectively. The route is from Yogyakarta to Palembang Malay to Mamuju then to Mandar.

4.1.2. Result of Yen’s K Shortest Path Algorithm

Yen’s K algorithm is different from the Dijkstra algorithm because there is a K value that can be adjusted as required. In this study, to measure the best algorithm performance, the K value used was 1, which meant that only one shortest path was returned. However, we show the results of using K = 4 for the first language pair used from Bali to Buginese in Table 4.

Table 4. Total Cost for the path from Bali to Buginese using Yen’s K Algorithm with K = 4.

Route	Intermediary Languages	Total Cost
1	PALEMBANG MALAY-REMUN-BOTTENG	196.10
2	PALEMBANG MALAY-TERNATE PASAR-BOTTENG	197.08
3	PALEMBANG MALAY-TAMUAN-BOTTENG	205.79
4	PALEMBANG MALAY-TERNATE PASAR-SANGIL	210.25

In the results shown in Table 4, four routes were selected according to the K value used. The first route with an index of 1 is from Bali to Remun via Palembang Malay and

from Remun to Buginese via Botteng; this route has a total cost of 196.1. The second route with the index of 2 is from Bali to Ternate Pasar via Palembang Malay and from Ternate Pasar to Buginese via Botteng; this route has a total cost of 197.08. The third route with the index of 3 is from Bali to Tamuan via Palembang Malay and from Tamuan to Buginese via Botteng; this route has a total cost of 205.79. The last route with the index of 4 is from Bali to Ternate Pasar via Palembang Malay and from Ternate Pasar to Buginese via Sangil; this route has a total cost of 210.25. The execution time required to obtain these four pathways in Yen’s K algorithm is 275 ms.

Next, we show the results of executing the Yen’s K algorithm for the three language pairs using the value of $K = 1$ to determine only the shortest path. Table 5 shows the results of the Neo4j Cypher projection from Bali to Buginese, Ambonese Malay to Karo Batak, and Yogyakarta to Mandar using the Yen’s K algorithm.

Table 5. Total Cost using Yen’s K Algorithm with $K = 1$.

Maximum Distance	Language Pair	Intermediary Languages	Total Cost
61	BALI-BUGINESE	PALEMBANG MALAY -REMUN -BOTTENG	196.10
60	AMBONESE MALAY -KARO BATAK	TERNATE PASAR	72.94
63	YOGYAKARTA -MANDAR	PALEMBANG MALAY -MAMUJU	147.69

The results from Bali to Buginese based on execution time and total cost calculated using the Yen’s K algorithm are 243 ms and 196.1, respectively. The route is from Bali to Remun via Palembang Malay and from Remun to Buginese via Botteng. With the distance property less than 60, the execution time for Ambonese Malay to Karo Batak is 301 ms with a total cost of 72.94; the route is from Ambonese Malay to Karo Batak via Ternate Pasar. The last language pair is obtained for the path from Yogyakarta to Mandar with a distance property less than 63. The execution time from Yogyakarta to Mandar is 292 ms with a total cost of 147.69; the route was from Yogyakarta to Palembang Malay to Mamuju and finally to Mandar.

4.1.3. Performance Comparison

Algorithm performance comparison includes the execution time and total cost. Figures 4 and 5 show a comparison of these two parameters.

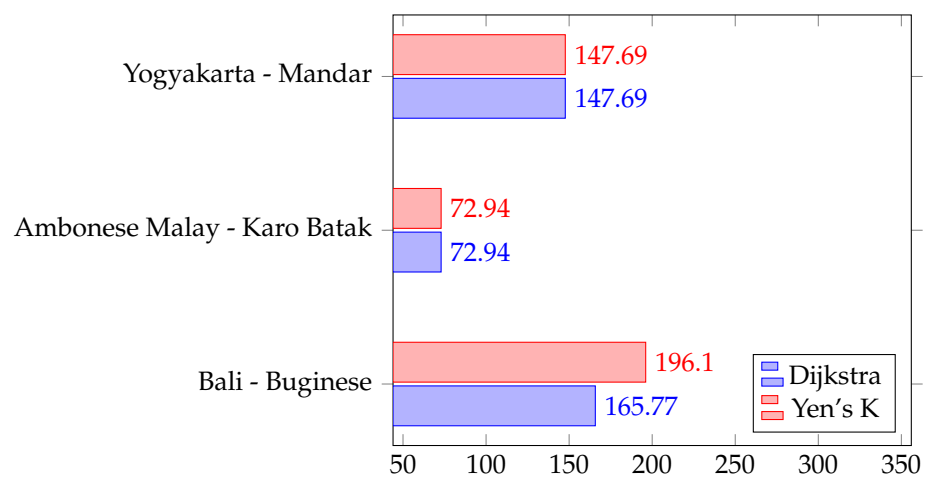


Figure 4. Performance comparison by total cost .

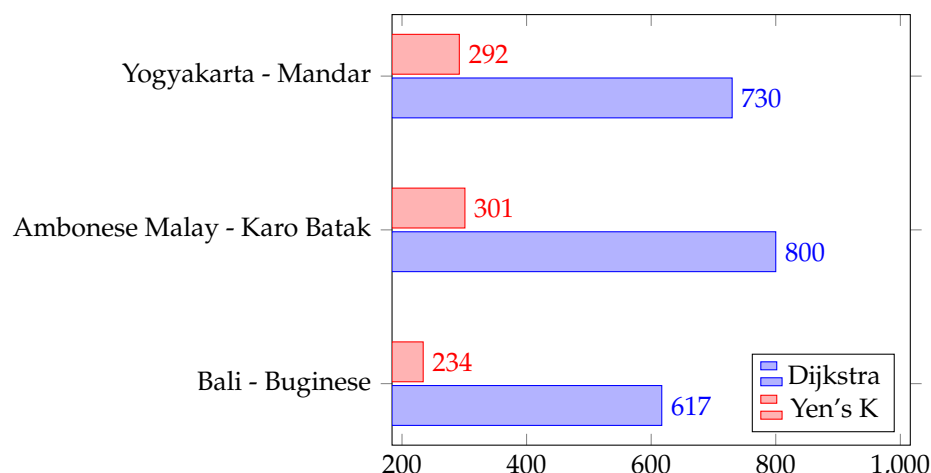


Figure 5. Performance comparison by execution time (ms).

Dijkstra and Yen's K give the same results for the total cost, except for Bali to Buginese, where Dijkstra outperformed Yen's K with 15.5% less cost. However, Yen's K algorithm has a faster execution time than the Dijkstra algorithm, with an average of 160% higher performance.

4.2. Mediator Selection from The Intermediary Languages

We present a case study where the intermediary closely related languages can be used as a guideline to find mediators who can help resolve the intertribal conflicts among Indonesian tribes. For this case study, we simulate the mediator selection process from the intermediary closely related languages obtained from BALI and BUGINESE as target languages. We obtained the language coordinates from the ASJP and further find the geographical location on the map based on the coordinates of each intermediary language, as shown in Table 6.

Table 6. Coordinates and language locations from Bali to Buginese obtained using Dijkstra.

Language	Coordinates	Location
BALI	8°20' S, 115°15' E	Buahan Kaja, Payangan, Kabupaten Gianyar, Bali
PALEMBANG MALAY	2°58'35.9" S, 104°46'30.8" E	Palembang, Lawang Kidul, Kec. Ilir Tim. II, Kota Palembang, Sumatera Selatan
EMBALOH	1°00'00.0" N, 112°00'00.0" E	Pulau Majang, Badau, Kabupaten Kapuas Hulu, Kalimantan Barat
BUGINESE	4°00'00.0" S, 120°00'00.0" E	Danau Buaya, Danau Tempe, Kabupaten Wajo, Sulawesi Selatan

The geographical distance between location can be calculated on Google Maps, as shown in Figure 6, where the total distance is 3256.34 km. Now that we know how to calculate the C_2 (average geographical distance between the mediator candidate's location with the target languages' locations), we need to use a tool to calculate C_1 .

To calculate C_1 (average language similarity between the mediator candidate's native language and the target languages), we implemented the Yen's K algorithm to calculate the shortest path between target languages and return the intermediary languages. We name the tool World Language Similarity Cluster (<https://world.langsphere.org>, accessed on 17 September 2022). Since the maximum distance is set to 61, as shown in Table 3, we can set the similarity as 39 to find the shortest path between BALI and BUGINESE. To obtain similarity between the languages, we can simply hover to the edges between any two nodes, as shown in Figure 7. Now that we also know how to calculate C_1 , the information from C_3 and C_4 can be integrated to calculate the overall cost using the weighted sum model, where the weight of each criteria can be defined by an expert in intertribal conflict resolution.

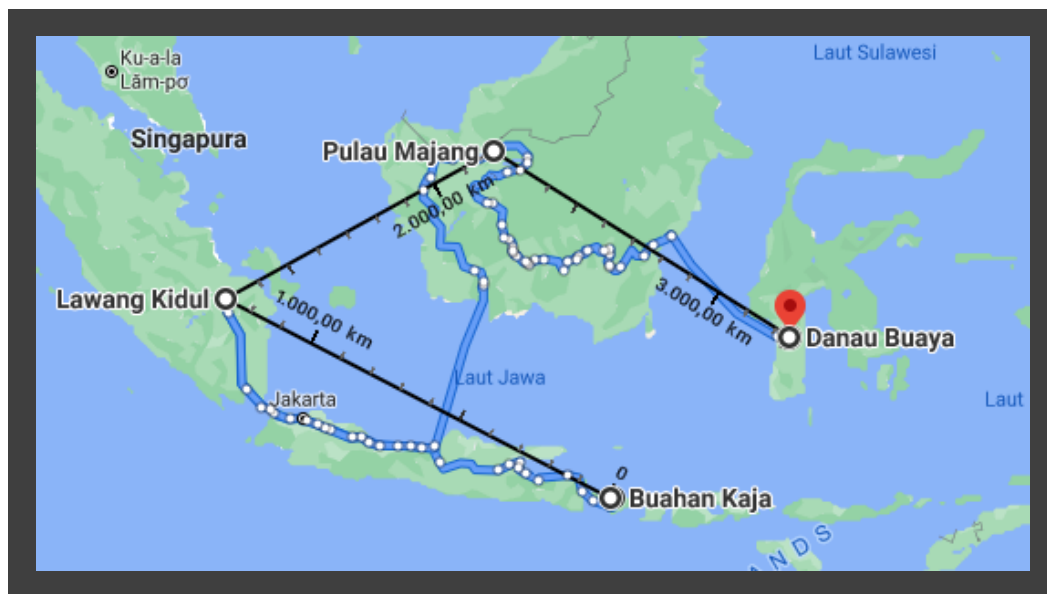


Figure 6. The distance between locations of Bali (Buahan Kaja) to Buginese (Danau Buaya) on maps using Dijkstra.

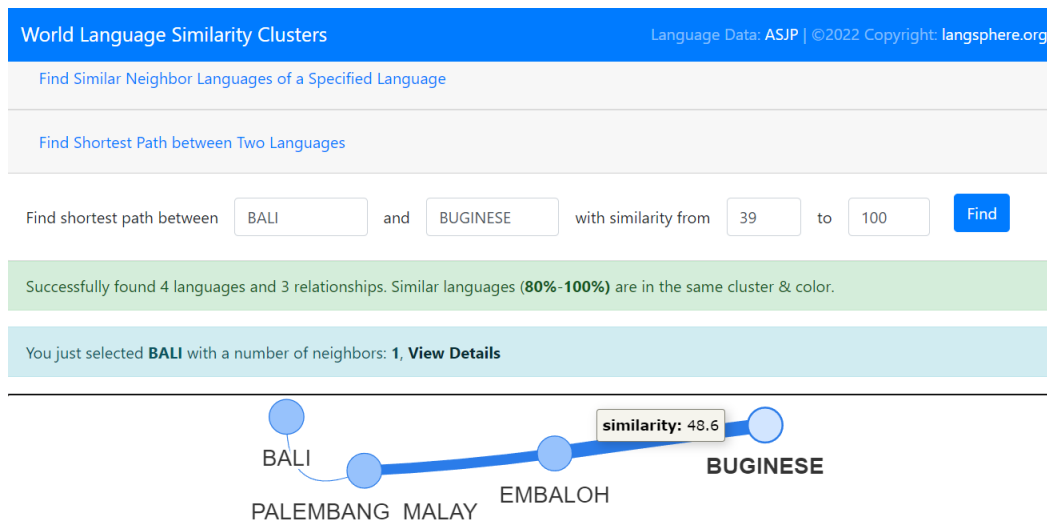


Figure 7. World Language Similarity Cluster (<https://world.langsphere.org>, accessed on 17 September 2022).

5. Conclusions

Our research findings prove that even though Dijkstra and Yen’s K algorithm have equal total cost for all the cases of Indonesian tribal languages, Yen’s K outperformed Dijkstra at searching for closely related intermediate languages, with an average of 160% higher performance on execution time. The selection of native speakers of the obtained intermediary languages as mediators is formalized as an optimization problem with four criteria: language similarity, geographical distance, background, and expected salary. We present a case study where the intermediary closely related languages can be used as a guideline to find mediators who can help resolve the intertribal conflicts among Indonesian tribes. To calculate the first criteria, we have implemented the Yen’s K algorithm to calculate the shortest path between target languages and returned the path via the intermediary languages. This implementation shows the potential use of the mediator selection model defined in this paper in various other roles, such as trader or salesman, politician’s spokesman, reporter or journalist, etc.

Author Contributions: Conceptualization, A.H.N., S.E.F. and R.S.; methodology, A.H.N., S.E.F. and R.S.; software, A.H.N. and S.E.F.; validation, W.M.; formal analysis, A.H.N. and R.S.; investigation, W.M., N.B.; resources, A.H.N.; data curation, A.H.N.; writing—original draft preparation, S.E.F.; writing—review and editing, A.H.N.; visualization, S.E.F. and W.M.; supervision, A.H.N. and R.S.; funding acquisition, A.H.N., N.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ASJP Automated Similarity Judgment Program

References

- Panggabean, S.R. *Conflict and Ethnic Peace in Indonesia [konflik dan Perdamaian Etnis di Indonesia]*; Pustaka Alvabet: Tangerang Selatan, Indonesia, 2018.
- Noor, A.F.; Sugito, S. Multicultural Education Based in Local Wisdom of Indonesia for Elementary Schools in the 21st Century. *J. Int. Soc. Stud.* **2019**, *9*, 94–106.
- Eko, B.S.; Putranto, H. Face Negotiation Strategy Based on Local Wisdom and Intercultural Competence to Promote Inter-ethnic Conflict Resolution: Case Study of Balinuraga, Lampung. *J. Intercult. Commun. Res.* **2021**, *50*, 506–540. [[CrossRef](#)]
- Dai, X. The development of interculturality and the management of intercultural conflict. In *Conflict Management and Intercultural Communication*; Routledge: London, UK, 2017; pp. 85–97.
- Weidmann, N.B. Geography as motivation and opportunity: Group concentration and ethnic conflict. *J. Confl. Resolut.* **2009**, *53*, 526–543. [[CrossRef](#)]
- Hernawan, W.; Pienrasmi, H.; Basri, H. The Implementation of Local Wisdom as an Ethnic Conflict Resolution. *Opción Rev. Cienc. Humanas Soc.* **2019**, *21*, 951–972.
- Dyck, K. Peacemakers in Action: Profiles of Religion in Conflict Resolution. *Peace Res.* **2007**, *39*, 150.
- Cohen, R. Language and conflict resolution: The limits of English. *Int. Stud. Rev.* **2001**, *3*, 25–51. [[CrossRef](#)]
- Na'im, A.; Syaputra, H. *Kewarganegaraan, Suku Bangsa, Agama dan Bahasa Sehari-Hari Penduduk Indonesia Hasil Sensus Penduduk 2010*; Badan Pusat Statistik: Jakarta, Indonesia, 2011.
- Mulyana, A. Hubungan Etnis Dalam Pendidikan Sejarah di Indonesia. Disajikan. Dalam. In Proceedings of the International Seminar on Ethnicity and Education, The Faculty of Education & Institute Research of Ethnicity Universiti Kebangsaan Malaysia, Kuala Lumpur, Malaysia, 28 March 2008.
- Nieke, N. Manajemen dan resolusi konflik dalam masyarakat. *J. Ilm. Pendidik. Lingkung. Dan Pembang.* **2011**, *12*, 51–60. [[CrossRef](#)]
- Sartika, R. Persepsi Mahasiswa Terhadap Konflik Dalam Pembelajaran Mata Kuliah Pendidikan Resolusi Konflik. *Edutech* **2017**, *16*, 85–97. [[CrossRef](#)]
- Bahari, Y. Model komunikasi lintas budaya dalam resolusi konflik berbasis Pranata Adat Melayu dan Madura di Kalimantan Barat. *J. Ilmu Komun.* **2014**, *6*, 1–12.
- Gooskens, C.; van Heuven, V.J.; Golubović, J.; Schüppert, A.; Swarte, F.; Voigt, S. Mutual intelligibility between closely related languages in Europe. *Int. J. Multiling.* **2018**, *15*, 169–193. [[CrossRef](#)]
- Abraham, R.G.; Chapelle, C.A. The meaning of cloze test scores: An item difficulty perspective. *Mod. Lang. J.* **1992**, *76*, 468–479. [[CrossRef](#)]
- Lehmann, W.P. *Historical Linguistics: An Introduction*; Routledge: London, UK; New York, NY, USA, 2013.
- Schepens, J.; Dijkstra, T.; Grootjen, F.; Van Heuven, W.J. Cross-language distributions of high frequency and phonetically similar cognates. *PLoS ONE* **2013**, *8*, e63006. [[CrossRef](#)]
- Dyen, I.; Kruskal, J.B.; Black, P. An Indoeuropean classification: A lexicostatistical experiment. *Trans. Am. Philos. Soc.* **1992**, *82*, iii–iv+1–132. [[CrossRef](#)]
- Nasution, A.H.; Murakami, Y.; Ishida, T. Constraint-based bilingual lexicon induction for closely related languages. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016; pp. 3291–3298.
- Nasution, A.H.; Murakami, Y.; Ishida, T. A generalized constraint approach to bilingual dictionary induction for low-resource language families. *ACM Trans. Asian Low-Resour. Lang. Inf. Process. (TALLIP)* **2017**, *17*, 9. [[CrossRef](#)]

21. Nasution, A.H.; Murakami, Y.; Ishida, T. Plan optimization for creating bilingual dictionaries of low-resource languages. In Proceedings of the 2017 International Conference on Culture and Computing (Culture and Computing), Kyoto, Japan, 10–12 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 35–41.
22. Nasution, A.H.; Murakami, Y.; Ishida, T. Designing a collaborative process to create bilingual dictionaries of Indonesian ethnic languages. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018.
23. Nasution, A.H.; Murakami, Y.; Ishida, T. Plan optimization to bilingual dictionary induction for low-resource language families. *Trans. Asian Low-Resour. Lang. Inf. Process.* **2021**, *20*, 29. [CrossRef]
24. Holman, E.W.; Wichmann, S.; Brown, C.H.; Velupillai, V.; Müller, A.; Bakker, D. Explorations in automated language classification. *Folia Linguist.* **2008**, *42*, 1–34. [CrossRef]
25. Brown, C.H.; Holman, E.W.; Wichmann, S.; Velupillai, V. Automated classification of the world's languages: A description of the method and preliminary results. *Lang. Typol. Univers.* **2008**, *61*, 285–308. [CrossRef]
26. Müller, A.; Wichmann, S.; Velupillai, V.; Brown, C.H.; Brown, P.; Sauppe, S.; Holman, E.W.; Bakker, D.; List, J.M.; Egorov, D.; et al. ASJP World Language Tree of Lexical Similarity: Version 3 (July 2010). Available online: <https://asjp.cldd.org/static/WorldLanguageTree-003.pdf> (accessed on 17 September 2022).
27. Brown, C.H.; Holman, E.W.; Wichmann, S. Sound correspondences in the world's languages. *Language* **2013**, *89*, 4–29. [CrossRef]
28. Bakker, D.; Müller, A.; Velupillai, V.; Wichmann, S.; Brown, C.H.; Brown, P.; Egorov, D.; Mailhammer, R.; Grant, A.; Holman, E.W. Adding typology to lexicostatistics: A combined approach to language classification. *Linguist. Typol.* **2009**, *13*, 161–189. [CrossRef]
29. Needham, M.; Hodler, A.E. *Graph Algorithms: Practical Examples in Apache Spark and Neo4j*; O'Reilly Media: Sebastopol, CA, USA, 2019.
30. Selim, H.; Zhan, J. Towards shortest path identification on large networks. *J. Big Data* **2016**, *3*, 10. [CrossRef]
31. Cui, X.; Shi, H. A*-based pathfinding in modern computer games. *Int. J. Comput. Sci. Netw. Secur.* **2011**, *11*, 125–130.
32. Magzhan, K.; Jani, H.M. A review and evaluations of shortest path algorithms. *Int. J. Sci. Technol. Res.* **2013**, *2*, 99–104.
33. Nasution, A.H.; Murakami, Y.; Ishida, T. Generating similarity cluster of Indonesian languages with semi-supervised clustering. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 531–538. [CrossRef]