THESIS

# THE ANTEATER ANALYSIS: A COMPARISON OF TRAVELING SALESMAN TOUR CONSTRUCTION METHODS AND THEIR GLOBAL FREQUENCIES

Submitted by

Shannon A. Ourada

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2022

Master's Committee:

Advisor: Darrell Whitley

Sudipto Ghosh
Benjamin Clegg

ABSTRACT

THE ANTEATER ANALYSIS: A COMPARISON OF TRAVELING SALESMAN TOUR CONSTRUCTION METHODS AND THEIR GLOBAL FREQUENCIES

For the Traveling Salesman Problem (TSP), many algorithms have been developed. These include heuristic solvers, such as nearest neighbors and ant colony optimization algorithms. In this work, the ATT48 and EIL101 instances are examined to better understand the difference between biased and unbiased methods of tour construction algorithms when combined with the 2-opt local search operator. First, a sample of tours are constructed. Then, we examine the frequencies of global edges of different sizes using n-grams. Using 2-opt as the tour improvement algorithm, we analyze randomly initialized local optima compared to nearest neighbors local optima as well as ant colony solutions with and without 2-opt. This comparison serves to better understand the nature of these different methods in their relation to the global optimum. We also provide some ways the algorithms may be adapted to take advantage of the global frequencies, particularly the ant colony optimization algorithm.

ACKNOWLEDGEMENTS

I could not have completed this work without the support and guidance of my advisor, Dr. Whitley. It was a privilege to be able to work with him as my advisor and to have him as a lecturer. It is rare to find someone with professionalism and a genuine warmth and care towards their students. I would also like to thank my parents for their endless love and support. Without them, this would not have been possible.

# DEDICATION

*I would like to dedicate this thesis to my brother, Ryan. He has always been there to challenge me to learn and to push myself further than I thought possible.*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

# Chapter 1

# Introduction

The Traveling Salesman Problem (TSP) has been researched widely in computer science and applied mathematics due to its complex nature and its ability to be translated to a variety of similar contexts. The TSP is a simple problem: find the shortest route to visit all of the cities in a given set exactly once and return to the starting point. However, the TSP is NP-Hard due to the number of combinations that can be chosen when exploring routes. With n cities in a symmetric problem, there can be ($n$ - 1)! / 2 possible solutions [1]. If we find ways to speed up finding the global optima or finding reasonably good solutions, we can apply these ideas to other NP-Hard problems.

The cities correspond to the vertices of a graph, G, and the edges are the connections between cities. Each solution to a TSP is a Hamiltonian Circuit, and a solution is typically represented as a permutation. A permutation is evaluated as shown in equation (1.1), where $\pi$ is the permutation and the distance between each edge in the Hamiltonian circuit is summed, including the edge from the last city back to the starting city.

$$\mathbf{f}(\pi) = d(\pi_1, \pi_n) + \sum_{i=1}^{n-1} d(\pi_i, \pi_{i+1}) \tag{1.1}$$

In this equation, *d(i,j)* is the distance corresponding to edge *e(i,j)* from the distance matrix. Typically, the *cost(i,j)* is the distance *d(i,j)*, but there are some variations that use other metrics for the cost.

In this work, we consider the symmetric TSP, though there are similarities between the symmetric and asymmetric TSP, and many optimization strategies can be used for both. In the symmetric instance of the TSP, distances between cities are the same in either direction. This is relevant when considering the distance matrix used to calculate distances between vertices as

well as the total route distance. It will also be relevant when we discuss patterns of edges when comparing local optima and the global optima.

There are exact algorithms for solving the TSP and approximate, or heuristic, algorithms for solving the TSP. Exact algorithms are guaranteed to find the optimal solution for TSP problems while approximate solutions find reasonably good solutions but are not guaranteed to find the optimal solutions. The global optimum for each TSP instance is the mathematically proven best solution. Local optima are defined with respect to their improvement operator and cannot be improved further with that operator but are not guaranteed to be the best possible solution. With exact solvers, the computation time needed to find the solution can be a drawback. On a 7397-city problem, it took approximately 3-4 years of CPU time to determine the exact solution on a large network of computers [1,2]. Improvements have been made since then, of course, but it is important to remember the complexity and computation demands of larger scale problems. Another example is the 13,509-city problem, which contains approximately $10^{50,000}$ possible tours; this problem took three months to solve with a cluster of 3 servers, 12 processors, and 32 PCs [1]. The current state-of-the-art exact solver is Concorde, which finds a globally optimal solution when it converges [3, 4]. In this work, we are using approximate algorithms. There are different types of approximate algorithms. One way to classify them is to split them into the following three categories: tour construction algorithms, tour improvement algorithms, and composite algorithms [1]. Tour construction algorithms build a solution by adding a new city at each step. Tour improvement algorithms take an already built tour and make changes to get better costs. Composite algorithms are a combination of the two.

We used different tour construction algorithms in this work to understand how they generate better solutions in terms of closeness to the global optimum. We examined two different tour construction algorithms: nearest neighbors and ant colony optimization. We compared both of these methods to tours that were randomly initialized. We used 2-opt for the tour improvement method to ensure that they were local optima. All local optima are defined with respect to a specific move operator. In our case, this is 2-opt. We used the standard 2-opt that

runs through the entire permutation and tests if swapping two sections of the tour will improve the overall cost. 2-opt continues to iterate over the tour until no further improvements can be made. As seen in Figure 1.1, 2-opt removes crossed edges by taking advantage of the triangle inequality: the sum of any two sides of a triangle is greater than or equal to the third side. A tour improved by 2-opt never has crossed edges.

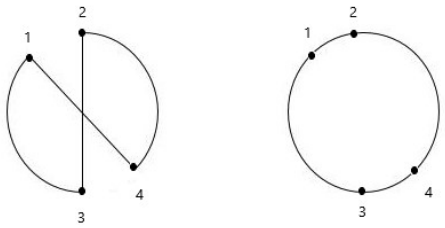



**Figure 1.1:** An example of a swap in 2-opt

In this work, we examine the frequency of edges in the global optimum for commonly studied TSP examples. We evaluated the frequency of n-grams, or permutation segments, of size 2, 3, 4, 5, and 7 for the ATT48 and EIL101. We compared these frequencies for solutions generated from randomly initialized tours, the nearest neighbors algorithm, and the ACO algorithm. Each of these tour construction methods was improved with 2-opt to generate the sample of 100 local optima for each, and we compared these to the ACO tours before they were improved with 2-opt as well.

We seek to determine whether ACO is better than other tour construction methods and to gain a better understanding of what the ACO algorithm is doing as it continues to explore more iterations. By comparing ACO and nearest neighbors to randomly generated paths, we evaluate the difference between biased and unbiased tour construction algorithms. Our goal is to determine whether it is advantageous to use a biased method of tour construction, like nearest neighbors or ACO, or if it is better to use an unbiased method like randomly generated tours. Our hypothesis is that unbiased methods will be better at sampling the global edges at higher frequencies. We believe that this comparison and evaluation of different methods of tour con-

struction can lead to improved heuristic solvers if we can use the frequency information to our advantage. There are also implications for future explorations of ACO and ways these frequencies can be used to enhance the ants' search space or used with other TSP algorithms.

# Chapter 2

# Background

In this section, ideas central to the understanding of the TSP and applications of our work will be discussed. This includes a brief overview of the tour construction methods and previous work related to the frequencies of globally optimal edges.

Using the different tour construction methods mentioned below, four different types of data samples were generated. Each sample had a total of 100 permutations.

Data Sample 1 is made up of Randomly Initialized Local Optima (RILO). These were generated by randomly initializing a permutation of the vertices, and then this candidate tour was improved using 2-opt until a local optimum is reached.

Data Sample 2 is composed of Nearest Neighbor Local Optima (NNLO). Permutations of the vertices were generated by applying one of several different greedy nearest neighbor selection methods. This sample included 20 permutations each of traditional nearest neighbors, nearest insertion, cheapest insertion, farthest insertion, and arbitrary insertion for a total of 100 permutations. After a permutation was constructed with a nearest neighbor method, the candidate tour was again improved using 2-opt until a local optimum was reached.

Data Sample 3 (ACO-S1) was generated by the ACO algorithm. These solutions were not improved, and they are not guaranteed to be local optima.

Data Sample 4 (ACO-LO) was the result of taking the tours generated from the ACO algorithm and improving them with 2-opt until a local optimum was reached for each permutation.

## 2.1 Tour Construction Algorithms

Aptly-named, tour construction algorithms are designed to find competitive solutions as they build each solution. As opposed to a randomly generated tour, tour construction algorithms build up a tour by adding a new city at each step [1]. Many of these algorithms employ a greedy strategy to determine which city to add next, such as choosing the closest city based

on the distance matrix. These algorithms can be much quicker to converge than starting with a random tour and improving upon it.

### 2.1.1 Nearest Neighbor and Insertion Algorithms

The nearest neighbor algorithm is a greedy tour construction algorithm. The algorithm works by choosing a city to start with and then selecting the nearest city of the cities that have not been visited yet. This continues until all of the cities have been visited and connects back to the first city. The greedy nature of this algorithm can create long edges towards the end of the tour, however. As the number of options decreases, the possibility of a large edge increases. This is why it is common to apply a tour improvement algorithm to results generated by nearest neighbors algorithms to ensure that the solutions are in fact local optima. This is what we do by applying 2-opt to the tours generated from nearest neighbors algorithms.

There are also variations on the original nearest neighbors algorithm to improve the generated solutions. One example of this is the repeated nearest neighbors algorithm. This variation starts the nearest neighbors algorithm from every vertex and compares all the results to select the best one [5]. There are also various insertion methods: nearest insertion, cheapest insertion, farthest insertion, and arbitrary insertion. Nearest insertion adds the nearest unvisited city to the current partial tour at every step [6]. Cheapest insertion adds the city that will add the least cost to the current total cost of the partial tour. Farthest insertion adds the farthest city from the current partial tour each time an unvisited city is added. Arbitrary insertion is similar to cheapest insertion, but instead of searching through all of the unvisited cities and selecting the cheapest, arbitrary insertion selects a random city not already on the partial tour and inserts it between neighboring cities on the tour in the cheapest way possible [7]. Arbitrary insertion can produce solutions with costs as much as $n$ times the optimal in the worst case for asymmetric TSP instances, but it produces solutions that are always better than twice the optimum for symmetric TSP instances [7]. Nearest insertion and cheapest insertion produce a tour that is no longer than twice the cost of the optimal solution as well [8]. Using a mixture of these different

types of nearest neighbors algorithms gave us a diverse initial population to compare after we ensured they were local optima by passing them through 2-opt.

### 2.1.2 Ant Colony Algorithm

While the ant colony algorithm has spawned a legion of similar swarm-based algorithms, it was novel when it was first published. Using nature as the inspiration, Dorigo and Gambardella explored the idea of using artificial ants to trace possible solutions, leaving behind a "pheromone" to highlight the paths that were already traversed [9]. In this way, the ants can choose to follow a path that the previous ants have already taken, leading to stronger solutions provided the pheromone is along the better path. Ant colony optimization (ACO) algorithms are population-based, meaning they use a population of ants and they find approximate solutions and continue to improve on those solutions as they run.

This idea was inspired by how real ants find the shortest path from their colony to a food source and how they can adapt if an obstacle creates the need for a new path [9]. When an obstacle is encountered, the shorter path will receive more pheromones so that a greater number of ants will choose the shorter path. This is compounded by how it takes longer to leave pheromones on the longer path, so the shorter path quickly gains more pheromones as more ants choose it and even more ants leave their pheromones along the shorter route. [9]. In this way, the ants continue to improve the routes traveled, and the goal is to find the shortest possible route.

In recent years, many swarm-based algorithms have gained popularity for their ability to generate diverse local optima and improve solutions. While the randomness of these algorithms can increase diversity within the generated local optima, they do not always perform better than other types of tour construction methods in terms of global edge frequencies, which is why we wanted to compare different tour construction algorithms. It may be possible to combine our work with edge frequencies with the ACO algorithm to improve the paths generated. Some applications of these algorithms are similar to the TSP, like the Vehicle Routing Problem,

but many were designed with the TSP in mind. The vehicle routing problem was originally called the "Truck Dispatching Problem", and instead of finding the optimal tour to hit all of the cities in an instance, it involves the optimal routing of a fleet of trucks between their hub and a large number of gas stations [10]. This idea was adapted into the Vehicle Routing Problem to make it more general for all kinds of logistics and transportation problems involving a set of customers and a centralized location for the delivery vehicles [11]. Instead of the ants serving as the salesman, the ants find the shortest routes for the vehicles to serve all of their customers. The choice of paths is probabilistic rather than deterministic.

The TSP is an interesting application of the ant colony algorithm. Instead of traversing a route from the anthill to a food source, artificial ants can travel to all of the cities in a TSP set and leave pheromones along their routes. As more ants try to find the shortest route, the pheromones will continue to build along the shorter paths. Part of the novelty of this algorithm is the pheromone and its evaporation rate. This can be used to randomize and add diversity to the routes selected by the ants. As the number of iterations increase, the ants' solutions should improve while maintaining a level of diversity in path choices. Otherwise, the ants could end up all along the same path too early, making better solutions impossible to reach.

We wanted to explore the original ant colony optimzation (ACO) algorithm. Our hypothesis is that ACO is not as effective at quickly finding globally optimal edges as the randomly initialized local optima or the nearest neighbor local optima. We also predict that there are ways to adapt the ACO algorithm to take advantage of the frequencies found in the local optima to find solutions closer to the global optimum. By using the frequencies in place of the pheromones, the ants could get much closer to reaching the global optimum at a faster rate.

With ACO, the ants traverse the graph, the set of cities, with the objective of finding the minimum distance between cities. This works by maintaining a matrix similar to a distance matrix to compute the minimum distance between nodes (cities). The parameters include the number of ants, the evaporation rate, the intensification, alpha, beta, an optional beta evaporation rate, and the probability to choose the best route [12]. The number of ants means that

for each iteration of the algorithm $n$ ants explore $n$ paths. The evaporation rate determines how rapidly the pheromone evaporates. This can be modified to allow more diversity in the paths chosen by the ants so that they do not all continue to explore the same path. The other benefit of the evaporation rate is that it can help the ants avoid being trapped in a local optimum, or suboptimal solution [13]. This algorithm also has an option to terminate early if a particular number of repeated best distances are found, and in our case, we set this to 20 for our initial comparison between the different tour construction methods. The intensification is the constant added to the best path at each iteration. Alpha is the weight of the pheromone, and beta is the weight of the heuristic (1/distance) [12]. Together alpha and beta affect how much influence the pheromones or the distance heuristic impact the ants' decisions. The heuristic can have an evaporation rate as well, but we did not include that in our work. The probability of choosing the best route determines how often the ant will choose the best route versus the route based on pheromones [12].

The pheromones are updated while constructing solutions and after all the ants have built a solution, meaning the pheromones are updated locally and globally [14]. The local update is to help ensure that the ants visit different paths rather than continuing to follow the same edges repeatedly. The global update encourages ants to remain close to the best solution found so far [13, 14]. The local pheromone update rule is

$$\tau_{ij}(t) \longleftarrow (1-\rho)\cdot\tau_{ij}(t) + \rho\cdot\tau_0 \tag{2.1}$$

where $\tau_{ij}$ is the amount of pheromone on the edge *(i,j)* at time $t$, $\rho$ is the pheromone evaporation rate, and $\tau_0$ is the initial pheromone value on all edges [14]. The global update is done once all the ants have explored routes for an iteration, and the best route is the one that is applied the intensification constant. The risk with the global pheromone update is that it can cause the ants to get stuck in a local optimum since the pheromone intensity will continue to be greatest on the current best path [15]. To remedy that, there is the "city select strategy," in which the next city visited is selected by comparing a prior probability $q_0$ and a random number $q$; if $q$

is greater an $q_0$, the ant will select the next city by the probability. The algorithm then maintains a probability matrix, a pheromone matrix, and a heuristic matrix as each change over the iterations of the algorithm.

## 2.2 TSP and Frequencies

In previous work with the TSP and edge frequency information, Varadarajan et al. examined two population-based heuristic solvers for the TSP: Edge Assembly Crossover (EAX) and the Lin-Kernigan-Helsgaun algorithm (LKH) [16]. Unlike construction algorithms that start at a single point and build up to a single solution, population-based solvers process a collection of individual sample solutions [17]. Each individual in the set, or the population, represents a possible solution to the problem, and the idea behind having a population is that the individual permutations will collaborate and compete to move the entire population toward better solutions [17]. Varadarajan et al. wanted to understand how these population-based solvers are effective and efficient, and the key to this understanding was examining the frequency of global edges in the population. They found that in the majority of the TSP local optima they examined, more than 73% of the global edges were represented in the initial population [16]. One of the major contributions of this work is the relationship between edge frequency and the global optimum: "relatively small populations that have been improved by applying 2-opt often contain all of the edges needed to reconstruct the global optimum" [16]. This means that it is highly likely that a randomly selected edge from a random local optimum is an edge that can also be found in the global optimum [16]. We hope to use this to continue to improve solutions found in optimization methods.

In Figure 2.1, the left column shows the population sizes required to find all of the global edges for cluster problems, the middle shows the minimum population sizes for printed circuit board (PCB) problems, and the right column shows the minimum population sizes required to find all of the global edges for the "city" problems [16]. If a particular population size was not enough to generate all of the global edges, the number was increased by a power of 2. For

example, the ATT532 required only 64 tours to have a set containing all of the edges of the global optimum.

| Cluster Problem | Pop Size | PCB Problem | Pop Size | City Problem | Pop Size |
|---|---|---|---|---|---|
| C1k.0 | 256 | pcb442 | 16 | att532 | 64 |
| C3k.0 | 256 | d657 | 256 | pr1002 | 128 |
| C3k.1 | 512 | pcb1173 | 256 | vm1084 | 512 |
| dsj1000 | 128 | d1291 | 128 | rl1304 | 64 |
| | | fl1400 | 128 | rl1323 | 256 |
| | | fl1577 | 4096 | nrw1379 | 128 |
| | | d1655 | 256 | vm1748 | 128 |
| | | u1817 | 128 | rl1889 | 128 |
| | | d2103 | 16384 | pr2392 | 128 |
| | | u2319 | 64 | fnl4461 | 256 |
| | | pcb3038 | 128 | | |

**Figure 2.1:** TSP Problem Instances and the Smallest Population Size Required to Contain All of the Globally Optimal Edges After Performing EAX based 2-opt on the Initial Population

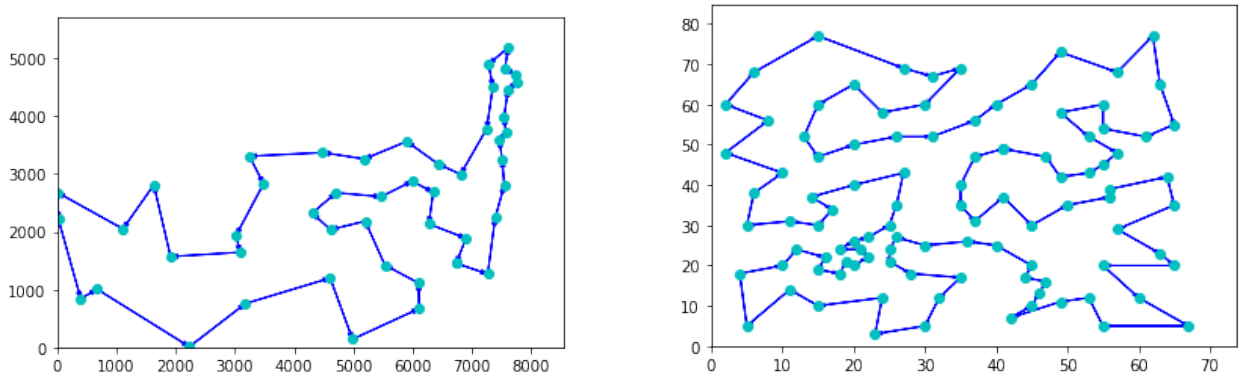
While Varadarajan et al. sought to answer the question of how many random 2-opt tours we need to get all of the global edges [16], we seek to explore more about the frequencies of the edges gathered in each set. We want to know how many high frequency edges we need to get all of the global edges. In our work, the frequencies of edges were obtained by passing the local optima and global optimum for each TSP instance through the n-gram package of Python3. An n-gram is a sequence of $n$ items, words, symbols, or tokens in a given text. Typically, n-grams are used in natural language processing (NLP) for specific tasks, such as word prediction, speech recognition, machine translation, and spelling correction [18]. However, n-grams have been used to compare features in other areas as well, like in software feature representation for network security [18]. We used n-grams to calculate the frequency of edges in the local optima and compare those edges with the global optimum. In this context, $n$ is the number of vertices, or cities, in a segment of a permutation of length $n$. We examined n-grams of size 2, 3, 4, 5, and 7. As $n$ increases, the constraints between different edges show interesting patterns as certain edges continue to appear at high frequencies.

# Chapter 3

# Methodology

## 3.1 The TSP Instances

Our approach involved comparing multiple well-known examples of the TSP, including the ATT48 and EIL101. We wanted to explore instances of different sizes to see if that impacted the patterns we noticed in the edge frequencies and if problems of different sizes changed performance levels for the various tour construction methods chosen. The ATT48 and EIL101 are both city problems, so the x and y coordinates in the graphs correspond to the coordinates of the cities in each set. The coordinates are used to construct distance matrices using the Euclidean distance formula to approximate the distances between cities. These TSP instances have a single global optimum, which made our comparison of globally optimal edges (global edges) and locally optimal edges (non-global edges) more straightforward.




**(a)** ATT48 Global Optimum

**(b)** EIL101 Global Optimum

**Figure 3.1:** The Global Optima for the ATT48 and EIL101

### 3.1.1 Gathering Local Optima

We also examined local optima generated in multiple ways. For each TSP instance, we looked at randomly initialized tours, nearest neighbors tours, and ACO generated paths. All of these were then passed through 2-opt to ensure they were in fact local optima with respect to 2-opt. We did, however, examine the ACO generated paths before and after running them through 2-opt to be able to examine the quality of the ACO generated paths. This generated our four data samples: randomly initalized local optima (RILO), nearest neighbors local optima (NNLO), the ACO sample 1 (ACO-S1), and ACO local optima (ACO-LO). We wanted to know if the ACO algorithm on its own (ACO-S1) can find global edges at high frequencies, or if 2-opt is necessary to generate more global edges.

With these local optima, we wanted to ensure that there was enough diversity to be able to see patterns in the edges that were in both the local optima and the global optimum. As noted in previous work [16], we do not need an overly large sample size to be able to represent all of the edges in the global optimum. Therefore, we decided to gather 100 local optima for the randomly generated set, the nearest neighbors set, and the ant colony set. All three sets were passed through 2-opt to ensure that no improvements could be made, meaning they were locally optimized. For the randomly generated set, we simply called on the random package in Python3 to select a permutation of size $n$ that matched the size of the TSP instance and included all of the vertices only once. For the nearest neighbors set, we used multiple different nearest neighbors algorithms to ensure a variety of paths: traditional nearest neighbors, nearest insertion, farthest insertion, cheapest insertion, and arbitrary insertion. There were 20 of each type for a total of 100 nearest neighbors paths. For the ant colony set, we ran the ant colony system code for a total of 300 iterations, and we collected the best paths for iterations from 200 to 300 to get a total of 100 paths. We used 100 ants with an evaporation rate of 0.1, an intensification rate of 2, alpha and beta were 1, and choose best at a rate of 0.1. An evaporation rate of 0.1 is often considered optimal [14].

## 3.2 Preparing N-Grams

To analyze the frequency of edges of varying sizes, the local optima were passed through the n-grams package in Python3. For each TSP instance and each type of generated local optima, we looked at n-grams of size 2, 3, 4, 5, and 7. An n-gram is a sequence of $n$ items, words, symbols, or tokens in a given text. In this context, the n-gram corresponds to the number of vertices, or cities, in an edge of the graph, so an n-gram of size 2, called a bigram, would be a connection between two cities. The larger the n-gram, the more constraints there are with which vertices can be next to each other. N-grams allowed us to analyze the frequency and optimality of patterns of different sizes.

To compare the n-grams with the global optima, the global optimum for each TSP instance was also ran through the n-gram code and then the dictionaries generated were compared by key. Using Python dictionaries, which store objects as key-value pairs each with a unique key, we were able to consolidate all of the edges and count the frequencies of global and non-global edges. Each dictionary had the edges sorted from smallest to largest in the keys to ensure that duplicates were merged and to make sure all edges that were in the global optimum and the local optima were found. The text had to be stripped and sorted before analyzing the n-grams, counting the frequency of the n-grams, and comparing the resulting n-grams to the global n-grams.

# Chapter 4

# Findings and Results

## 4.1 ATT48 Frequencies

The ATT48 is a 48-city problem. After gathering all of the local optima for each type of tour construction, each of the 100 local optima were passed through the n-gram code, sorted according to the smallest first edge, and compared to the n-grams of the same size in the global optimum for the ATT48. These were then saved in Microsoft Excel workbooks to sort and analyze the results. For each size of n-gram, the results were sorted from highest to lowest frequency, and the global and non-global edges were color coded. In the tables and figures below, the results are shown. The tables include the number of global edges found by each set of local optima, the percentage of global frequencies, and sorted n-grams of different sizes. In the plots, the frequency vs number of n-grams are shown in histograms. All of these results provide ways to compare randomly initialized local optima (RILO), nearest neighbors local optima (NNLO), and ant colony tours before (ACO-S1) and after 2-opt (ACO-LO).

Table 4.1 shows how many of the global edges, of a total of 48 for the ATT48, were represented for each type of tour construction method. Table 4.2 shows the percentage of global edges compared to the number of unique edges generated by each tour construction method. For this second table, the global frequencies were counted and divided by the total number of frequencies (global and non-global). These tables demonstrate the effectiveness of each method at sampling the global optimum. However, they do not show the full picture.

In Table 4.3, the top 20 and bottom 20 ranked global edges for the bigrams are shown. Edges were sorted from the highest to lowest frequency, so the lower the rank, the higher the frequency of the edge. RILO, NNLO, and ACO-LO found the first 20 global edges in the top 21 frequencies, while ACO-S1 found the top 20 within the top 30 frequencies. At the bottom, it took RILO 91 of the top frequencies to get all of the global edges, followed by ACO-LO and NNLO each with 111,

**Table 4.1:** Table of Global Edges Found For Each Tour Construction for the ATT48.
This shows whether each method found all of the global edges of each size within the 100 sampled tours.

| ATT48 Global Edges Count | | | | |
|---|---|---|---|---|
| n-grams | RILO | NNLO | ACO-S1 | ACO-LO |
| 2 | 48 | 48 | 42 | 48 |
| 3 | 48 | 48 | 35 | 48 |
| 4 | 47 | 47 | 30 | 48 |
| 5 | 46 | 44 | 22 | 48 |
| 7 | 45 | 43 | 12 | 48 |

**Table 4.2:** Table of the Percent of Global Edges Out of the Total Frequency for Each Tour.
This shows the rate at which the global edges were present within all of the edges found within the 100 sampled tours for each construction method.

| ATT48 Percentage of Global N-Grams | | | | |
|---|---|---|---|---|
| n-grams | RILO | NNLO | ACO-S1 | ACO-LO |
| 2 | 70.06% | 67.42% | 53.01% | 70.51% |
| 3 | 49.28% | 48.27% | 21.21% | 48.94% |
| 4 | 41.98% | 33.39% | 10.58% | 38.41% |
| 5 | 31.77% | 26.14% | 4.11% | 26.98% |
| 7 | 25.78% | 17.00% | 1.16% | 18.53% |

and ACO-S1 needed 296 of the top frequency bigrams to get all of the global edges it contained. ACO-S1 only found 42 of the 48 global bigrams.

Similarly, in table 4.4, the top 20 and bottom 20 highest frequency global trigrams are depicted. Again, RILO found the top 20 global edges within the top 21 frequencies, followed by ACO-LO at 26, NNLO at 28, and ACO-S1 at 58. Note that ACO-S1 did not have a global edge until the top 11th frequency. In the bottom 20, RILO found all of the global edges within 212 top frequencies, NNLO within 254, ACO-LO within 425, and ACO-S1 by the top 829th frequency. ACO-S1 only found 35 of the 48 global trigrams as noted in table 4.1, so there are some trigrams repeated in the top and bottom 20.

In Table 4.5, the top 20 highest frequency and bottom 20 highest frequency global 5-grams for RILO, NNLO, and ACO-LO are shown. ACO-S1 is in the Appendix in Table A.1. ACO-S1 only found 22 of the 48 global 5-grams, and the top 20 frequencies of the 22 found were in the top

frequencies of the range of 35 to 1699. RILO were able to find the top 20 by the 27th highest frequency, NNLO by the 36th highest frequency, and ACO-LO by the top 46th. To find all of the global 5-grams each contains, RILO found 46 of the 48 by the 270th frequency, NNLO found 44 by the 1027th frequency, and ACO-LO found all 48 by the 922nd. While RILO did not find all of the global edges, much less of the 5-gram table needed to be traversed before getting all of the possible global edges, which would make reconstructing the global path from this set much more efficient than the other biased tour construction methods.

In Table 4.6 and Table 4.7, the frequencies of the top 40 bigrams and trigrams are arranged according to frequency, and the highlighted edges are global edges. For the bigrams, RILO and NNLO both featured only global edges in the top 25 while ACO-S1 and ACO-LO both had some non-global edges in the top 25. Of course, the top 40 will not include all of the global edges for any of the methods. Similarly, the trigrams in Table 4.7 are global except for one in RILO while NNLO and ACO-LO tours had some non-global edges, and ACO-S1 had mostly non-global edges in the top 40 trigrams. Then, there is Table 4.8, which features the top 40 n-grams of size 5 for the ATT48. ACO-S1 is in the Appendix in Table A.2. The top 40 highest frequency 5-grams in the ACO-S1 were mainly non-global edges. In fact, the first global 5-gram is the 35th rank with a frequency of 20.

Following the tables, the plots for the bigram frequencies are depicted in figures 4.1 and 4.2. In each plot, the global bigrams are blue and the non-global edges are red. The data plotted is cut off at 10 edges after the last global edge found. The frequencies are arranged highest to lowest so that the frequencies and their optimality can quickly be compared for each type of tour construction. RILO and NNLO both have large sections of high global frequencies, while the ACO-S1 plot has patches of red fairly early. This demonstrates that RILO not only do an excellent job at finding the globally optimal edges, but they also find many of them at high frequencies.

In Figure 4.3, the top 100 bigrams for RILO and NNLO are shown while the top bigrams for ACO-S1 and ACO-LO are displayed in Figure 4.4. The global edges are blue and the non-global

edges are red. Notice that the NNLO graph is missing two of the global edges while the graph of RILO has all of the global bigrams represented in the top 100. ACO-S1 is missing several of the global bigrams, and as noted in Table 4.1, the ACO-S1 tours only contained 42 of the global bigrams. ACO-LO was able to gather all of the global bigrams, but not within 100 ranks, so two of the global bigrams are missing from the plot. As seen in Table 4.3, ACO-LO and NNLO each needed 111 of the top frequencies to get all of the global edges while RILO obtained them all within 91 of the top frequencies. We see many of the same non-global edges represented, though there are some differences when comparing the four graphs.

**Table 4.3:** Table of Top 20 and Bottom 20 Global Bigrams for the ATT48
Edge (16,26) is shown in the edge column as | 16 | 26 |. Edges are sorted from highest to lowest frequency.

| **ATT48 Top 20, Bottom 20 Global Bigrams** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Random** | | | **NN** | | | **ACO** | | | **ACO 2-opt** | | |
| **rank** | **edge** | | **rank** | **edge** | | **rank** | **edge** | | **rank** | **edge** | |
| 1 | 16 | 26 | 1 | 16 | 42 | 1 | 31 | 38 | 1 | 5 | 36 |
| 2 | 16 | 42 | 2 | 18 | 26 | 2 | 9 | 23 | 2 | 18 | 26 |
| 3 | 18 | 26 | 3 | 18 | 36 | 3 | 34 | 44 | 3 | 18 | 36 |
| 4 | 18 | 36 | 4 | 5 | 36 | 4 | 3 | 25 | 4 | 16 | 26 |
| 5 | 3 | 25 | 5 | 3 | 25 | 5 | 0 | 7 | 5 | 16 | 42 |
| 6 | 3 | 34 | 6 | 34 | 44 | 10 | 1 | 28 | 6 | 9 | 44 |
| 7 | 34 | 44 | 7 | 16 | 26 | 12 | 4 | 47 | 7 | 34 | 44 |
| 8 | 5 | 36 | 8 | 15 | 21 | 13 | 15 | 40 | 8 | 3 | 34 |
| 9 | 0 | 7 | 9 | 4 | 47 | 15 | 33 | 40 | 9 | 3 | 25 |
| 10 | 15 | 21 | 10 | 0 | 7 | 16 | 16 | 42 | 10 | 1 | 28 |
| 11 | 30 | 43 | 11 | 3 | 34 | 17 | 32 | 45 | 11 | 17 | 43 |
| 12 | 29 | 42 | 12 | 31 | 38 | 18 | 15 | 21 | 12 | 6 | 17 |
| 13 | 5 | 27 | 13 | 30 | 43 | 19 | 18 | 36 | 13 | 31 | 38 |
| 14 | 1 | 28 | 14 | 29 | 42 | 20 | 6 | 17 | 14 | 33 | 40 |
| 15 | 31 | 38 | 15 | 5 | 27 | 22 | 17 | 43 | 15 | 0 | 7 |
| 16 | 33 | 40 | 16 | 30 | 37 | 24 | 11 | 14 | 16 | 15 | 21 |
| 17 | 4 | 47 | 17 | 32 | 45 | 25 | 19 | 46 | 17 | 5 | 27 |
| 18 | 30 | 37 | 18 | 33 | 40 | 27 | 13 | 24 | 19 | 29 | 42 |
| 19 | 17 | 43 | 19 | 13 | 24 | 28 | 29 | 42 | 20 | 30 | 43 |
| 20 | 6 | 17 | 20 | 1 | 28 | 30 | 2 | 22 | 21 | 15 | 40 |
| | | | | | | | | | | | |
| 31 | 9 | 23 | 33 | 38 | 47 | 35 | 30 | 43 | 35 | 14 | 39 |
| 32 | 20 | 46 | 34 | 19 | 46 | 36 | 19 | 32 | 36 | 35 | 45 |
| 34 | 19 | 32 | 36 | 20 | 46 | 37 | 14 | 39 | 37 | 9 | 23 |
| 35 | 11 | 14 | 38 | 12 | 24 | 41 | 16 | 26 | 38 | 2 | 22 |
| 36 | 35 | 45 | 39 | 19 | 32 | 42 | 13 | 22 | 39 | 19 | 32 |
| 37 | 1 | 25 | 40 | 10 | 11 | 45 | 5 | 27 | 40 | 10 | 11 |
| 39 | 10 | 11 | 41 | 14 | 39 | 47 | 12 | 24 | 41 | 23 | 41 |
| 42 | 23 | 41 | 44 | 20 | 31 | 49 | 5 | 36 | 42 | 1 | 25 |
| 43 | 15 | 40 | 45 | 8 | 39 | 50 | 20 | 46 | 43 | 20 | 46 |
| 45 | 38 | 47 | 48 | 35 | 45 | 51 | 10 | 11 | 46 | 12 | 24 |
| 46 | 13 | 22 | 49 | 1 | 25 | 53 | 6 | 27 | 47 | 2 | 21 |
| 47 | 12 | 24 | 51 | 13 | 22 | 56 | 29 | 35 | 51 | 13 | 22 |
| 48 | 8 | 39 | 55 | 7 | 37 | 57 | 2 | 21 | 59 | 38 | 47 |
| 49 | 20 | 31 | 59 | 4 | 41 | 70 | 8 | 39 | 60 | 20 | 31 |
| 58 | 2 | 22 | 60 | 23 | 41 | 83 | 10 | 12 | 65 | 8 | 39 |
| 59 | 4 | 41 | 65 | 2 | 22 | 89 | 7 | 37 | 70 | 7 | 37 |
| 68 | 7 | 37 | 66 | 15 | 40 | 111 | 0 | 8 | 76 | 10 | 12 |
| 74 | 10 | 12 | 71 | 0 | 8 | 120 | 35 | 45 | 80 | 0 | 8 |
| 82 | 28 | 33 | 102 | 10 | 12 | 251 | 20 | 31 | 109 | 4 | 41 |
| 91 | 0 | 8 | 111 | 28 | 33 | 296 | 1 | 25 | 111 | 28 | 33 |

**Table 4.4:** Table of Top 20 and Bottom 20 Global Trigrams for the ATT48
Edge (2, 34, 44) is shown in the edge column as | 3 | 34 | 44 |. Edges are sorted from highest to lowest frequency.

| **ATT48 Top 20, Bottom 20 Global Trigrams** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Random** | | | | **NN** | | | | **ACO** | | | | **ACO 2-opt** | | | |
| **rank** | **edge** | | | **rank** | **edge** | | | **rank** | **edge** | | | **rank** | **edge** | | |
| 1 | 3 | 34 | 44 | 1 | 26 | 16 | 42 | 11 | 15 | 40 | 33 | 1 | 9 | 44 | 34 |
| 2 | 5 | 36 | 18 | 2 | 5 | 36 | 18 | 13 | 21 | 15 | 40 | 2 | 3 | 34 | 44 |
| 3 | 26 | 18 | 36 | 3 | 16 | 26 | 18 | 17 | 6 | 17 | 43 | 3 | 25 | 3 | 34 |
| 4 | 26 | 16 | 42 | 4 | 25 | 3 | 34 | 18 | 16 | 42 | 29 | 4 | 6 | 17 | 43 |
| 5 | 25 | 3 | 34 | 5 | 3 | 34 | 44 | 23 | 26 | 18 | 36 | 5 | 5 | 36 | 18 |
| 6 | 16 | 26 | 18 | 6 | 26 | 18 | 36 | 25 | 19 | 32 | 45 | 6 | 26 | 18 | 36 |
| 7 | 16 | 42 | 29 | 7 | 16 | 42 | 29 | 26 | 17 | 43 | 30 | 7 | 26 | 16 | 42 |
| 8 | 27 | 5 | 36 | 8 | 27 | 5 | 36 | 27 | 32 | 19 | 46 | 8 | 16 | 42 | 29 |
| 9 | 6 | 17 | 43 | 9 | 6 | 17 | 43 | 30 | 26 | 16 | 42 | 10 | 16 | 26 | 18 |
| 10 | 35 | 29 | 42 | 10 | 37 | 30 | 43 | 31 | 11 | 14 | 39 | 11 | 17 | 43 | 30 |
| 11 | 37 | 30 | 43 | 11 | 2 | 21 | 15 | 34 | 22 | 13 | 24 | 12 | 15 | 40 | 33 |
| 12 | 5 | 27 | 6 | 12 | 17 | 43 | 30 | 36 | 5 | 36 | 18 | 13 | 21 | 15 | 40 |
| 13 | 9 | 44 | 34 | 13 | 9 | 44 | 34 | 37 | 10 | 11 | 14 | 14 | 27 | 5 | 36 |
| 14 | 17 | 43 | 30 | 14 | 23 | 9 | 44 | 41 | 37 | 30 | 43 | 15 | 37 | 30 | 43 |
| 16 | 2 | 21 | 15 | 15 | 35 | 29 | 42 | 42 | 19 | 46 | 20 | 18 | 5 | 27 | 6 |
| 17 | 17 | 6 | 27 | 16 | 5 | 27 | 6 | 46 | 17 | 6 | 27 | 19 | 35 | 29 | 42 |
| 18 | 23 | 9 | 44 | 18 | 4 | 47 | 38 | 47 | 2 | 22 | 13 | 23 | 17 | 6 | 27 |
| 19 | 1 | 25 | 3 | 21 | 31 | 38 | 47 | 49 | 12 | 24 | 13 | 24 | 29 | 35 | 45 |
| 20 | 29 | 35 | 45 | 24 | 17 | 6 | 27 | 55 | 16 | 26 | 18 | 25 | 23 | 9 | 44 |
| 21 | 25 | 1 | 28 | 28 | 20 | 31 | 38 | 58 | 35 | 29 | 42 | 26 | 25 | 1 | 28 |
| | | | | | | | | | | | | | | | |
| 37 | 9 | 23 | 41 | 46 | 9 | 23 | 41 | 46 | 17 | 6 | 27 | 47 | 22 | 13 | 24 |
| 38 | 4 | 47 | 38 | 47 | 0 | 7 | 37 | 47 | 2 | 22 | 13 | 52 | 12 | 24 | 13 |
| 39 | 10 | 11 | 14 | 49 | 10 | 11 | 14 | 49 | 12 | 24 | 13 | 55 | 21 | 2 | 22 |
| 40 | 20 | 31 | 38 | 53 | 22 | 13 | 24 | 55 | 16 | 26 | 18 | 60 | 20 | 31 | 38 |
| 42 | 22 | 13 | 24 | 54 | 11 | 14 | 39 | 58 | 35 | 29 | 42 | 61 | 31 | 38 | 47 |
| 48 | 41 | 4 | 47 | 66 | 4 | 41 | 23 | 60 | 2 | 21 | 15 | 65 | 9 | 23 | 41 |
| 51 | 8 | 39 | 14 | 67 | 31 | 20 | 46 | 64 | 5 | 27 | 6 | 79 | 0 | 7 | 37 |
| 52 | 12 | 24 | 13 | 68 | 32 | 19 | 46 | 71 | 27 | 5 | 36 | 85 | 2 | 22 | 13 |
| 57 | 4 | 41 | 23 | 70 | 21 | 15 | 40 | 72 | 21 | 2 | 22 | 98 | 8 | 39 | 14 |
| 65 | 31 | 20 | 46 | 72 | 7 | 37 | 30 | 111 | 0 | 7 | 37 | 99 | 31 | 20 | 46 |
| 66 | 0 | 7 | 37 | 78 | 8 | 39 | 14 | 120 | 8 | 39 | 14 | 111 | 7 | 0 | 8 |
| 70 | 21 | 2 | 22 | 81 | 7 | 0 | 8 | 141 | 7 | 0 | 8 | 113 | 7 | 37 | 30 |
| 90 | 28 | 33 | 40 | 93 | 0 | 8 | 39 | 159 | 7 | 37 | 30 | 116 | 10 | 12 | 24 |
| 99 | 1 | 28 | 33 | 97 | 21 | 2 | 22 | 169 | 0 | 8 | 39 | 118 | 0 | 8 | 39 |
| 116 | 7 | 37 | 30 | 107 | 15 | 40 | 33 | 259 | 32 | 45 | 35 | 123 | 11 | 10 | 12 |
| 127 | 0 | 8 | 39 | 172 | 10 | 12 | 24 | 545 | 20 | 31 | 38 | 142 | 4 | 47 | 38 |
| 135 | 7 | 0 | 8 | 175 | 2 | 22 | 13 | 562 | 29 | 35 | 45 | 198 | 4 | 41 | 23 |
| 155 | 10 | 12 | 24 | 212 | 28 | 33 | 40 | 731 | 1 | 25 | 3 | 204 | 28 | 33 | 40 |
| 171 | 11 | 10 | 12 | 230 | 1 | 28 | 33 | 828 | 10 | 12 | 24 | 376 | 1 | 28 | 33 |
| 212 | 2 | 22 | 13 | 254 | 11 | 10 | 12 | 829 | 11 | 10 | 12 | 425 | 41 | 4 | 47 |

**Table 4.5:** Table of Top 20 and Bottom 20 Global 5-grams for the ATT48
Edge (5, 36, 18, 26, 16) is shown in the edge column as | 5 | 36 | 18 | 26 | 16 |. Edges are sorted from highest to lowest frequency.

| **ATT48 Top 20, Bottom 20 Global 5-grams** | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Random** | | | | | | **NN** | | | | | | **ACO 2-opt** | | | | | |
| **rank** | **edge** | | | | | **rank** | **edge** | | | | | **rank** | **edge** | | | | |
| 1 | 5 | 36 | 18 | 26 | 16 | 1 | 18 | 26 | 16 | 42 | 29 | 1 | 9 | 44 | 34 | 3 | 25 |
| 2 | 26 | 18 | 36 | 5 | 27 | 2 | 36 | 18 | 26 | 16 | 42 | 2 | 5 | 36 | 18 | 26 | 16 |
| 3 | 36 | 18 | 26 | 16 | 42 | 3 | 5 | 36 | 18 | 26 | 16 | 3 | 36 | 18 | 26 | 16 | 42 |
| 4 | 18 | 26 | 16 | 42 | 29 | 4 | 26 | 18 | 36 | 5 | 27 | 4 | 18 | 26 | 16 | 42 | 29 |
| 5 | 26 | 16 | 42 | 29 | 35 | 5 | 3 | 34 | 44 | 9 | 23 | 5 | 26 | 18 | 36 | 5 | 27 |
| 6 | 6 | 27 | 5 | 36 | 18 | 6 | 9 | 44 | 34 | 3 | 25 | 6 | 6 | 17 | 43 | 30 | 37 |
| 7 | 9 | 44 | 34 | 3 | 25 | 7 | 26 | 16 | 42 | 29 | 35 | 7 | 26 | 16 | 42 | 29 | 35 |
| 8 | 6 | 17 | 43 | 30 | 37 | 8 | 6 | 27 | 5 | 36 | 18 | 9 | 27 | 6 | 17 | 43 | 30 |
| 9 | 5 | 27 | 6 | 17 | 43 | 9 | 6 | 17 | 43 | 30 | 37 | 10 | 5 | 27 | 6 | 17 | 43 |
| 10 | 3 | 34 | 44 | 9 | 23 | 12 | 27 | 6 | 17 | 43 | 30 | 12 | 6 | 27 | 5 | 36 | 18 |
| 11 | 27 | 6 | 17 | 43 | 30 | 13 | 5 | 27 | 6 | 17 | 43 | 14 | 17 | 6 | 27 | 5 | 36 |
| 12 | 17 | 6 | 27 | 5 | 36 | 14 | 17 | 6 | 27 | 5 | 36 | 15 | 16 | 42 | 29 | 35 | 45 |
| 13 | 16 | 42 | 29 | 35 | 45 | 17 | 4 | 47 | 38 | 31 | 20 | 16 | 3 | 34 | 44 | 9 | 23 |
| 14 | 1 | 25 | 3 | 34 | 44 | 18 | 16 | 42 | 29 | 35 | 45 | 17 | 1 | 25 | 3 | 34 | 44 |
| 15 | 28 | 1 | 25 | 3 | 34 | 19 | 1 | 25 | 3 | 34 | 44 | 18 | 28 | 1 | 25 | 3 | 34 |
| 16 | 32 | 45 | 35 | 29 | 42 | 22 | 32 | 45 | 35 | 29 | 42 | 20 | 32 | 45 | 35 | 29 | 42 |
| 21 | 19 | 32 | 45 | 35 | 29 | 25 | 28 | 1 | 25 | 3 | 34 | 30 | 19 | 32 | 45 | 35 | 29 |
| 23 | 34 | 44 | 9 | 23 | 41 | 28 | 34 | 44 | 9 | 23 | 41 | 35 | 2 | 21 | 15 | 40 | 33 |
| 26 | 35 | 45 | 32 | 19 | 46 | 30 | 19 | 32 | 45 | 35 | 29 | 38 | 35 | 45 | 32 | 19 | 46 |
| 27 | 4 | 41 | 23 | 9 | 44 | 36 | 9 | 23 | 41 | 4 | 47 | 46 | 22 | 2 | 21 | 15 | 40 |
| | | | | | | | | | | | | | | | | | |
| 55 | 46 | 20 | 31 | 38 | 47 | 64 | 35 | 45 | 32 | 19 | 46 | 154 | 37 | 7 | 0 | 8 | 39 |
| 71 | 1 | 28 | 33 | 40 | 15 | 65 | 37 | 7 | 0 | 8 | 39 | 155 | 8 | 0 | 7 | 37 | 30 |
| 73 | 19 | 46 | 20 | 31 | 38 | 66 | 8 | 0 | 7 | 37 | 30 | 159 | 31 | 20 | 46 | 19 | 32 |
| 74 | 21 | 15 | 40 | 33 | 28 | 101 | 0 | 7 | 37 | 30 | 43 | 162 | 0 | 8 | 39 | 14 | 11 |
| 75 | 22 | 2 | 21 | 15 | 40 | 129 | 19 | 46 | 20 | 31 | 38 | 169 | 0 | 7 | 37 | 30 | 43 |
| 99 | 3 | 25 | 1 | 28 | 33 | 142 | 7 | 37 | 30 | 43 | 17 | 195 | 11 | 10 | 12 | 24 | 13 |
| 108 | 25 | 1 | 28 | 33 | 40 | 204 | 20 | 46 | 19 | 32 | 45 | 209 | 7 | 0 | 8 | 39 | 14 |
| 110 | 0 | 7 | 37 | 30 | 43 | 216 | 2 | 21 | 15 | 40 | 33 | 238 | 13 | 22 | 2 | 21 | 15 |
| 133 | 8 | 0 | 7 | 37 | 30 | 266 | 7 | 0 | 8 | 39 | 14 | 239 | 21 | 2 | 22 | 13 | 24 |
| 134 | 8 | 39 | 14 | 11 | 10 | 267 | 0 | 8 | 39 | 14 | 11 | 243 | 14 | 11 | 10 | 12 | 24 |
| 149 | 31 | 20 | 46 | 19 | 32 | 268 | 8 | 39 | 14 | 11 | 10 | 322 | 12 | 10 | 11 | 14 | 39 |
| 151 | 37 | 7 | 0 | 8 | 39 | 315 | 22 | 2 | 21 | 15 | 40 | 326 | 4 | 47 | 38 | 31 | 20 |
| 153 | 0 | 8 | 39 | 14 | 11 | 363 | 21 | 15 | 40 | 33 | 28 | 327 | 4 | 41 | 23 | 9 | 44 |
| 162 | 7 | 37 | 30 | 43 | 17 | 437 | 1 | 28 | 33 | 40 | 15 | 787 | 3 | 25 | 1 | 28 | 33 |
| 188 | 7 | 0 | 8 | 39 | 14 | 503 | 10 | 12 | 24 | 13 | 22 | 788 | 25 | 1 | 28 | 33 | 40 |
| 193 | 12 | 10 | 11 | 14 | 39 | 504 | 11 | 10 | 12 | 24 | 13 | 789 | 1 | 28 | 33 | 40 | 15 |
| 225 | 10 | 12 | 24 | 13 | 22 | 514 | 31 | 20 | 46 | 19 | 32 | 917 | 21 | 15 | 40 | 33 | 28 |
| 227 | 11 | 10 | 12 | 24 | 13 | 937 | 13 | 22 | 2 | 21 | 15 | 920 | 31 | 38 | 47 | 4 | 41 |
| 234 | 14 | 11 | 10 | 12 | 24 | 1026 | 3 | 25 | 1 | 28 | 33 | 921 | 23 | 41 | 4 | 47 | 38 |
| 270 | 2 | 22 | 13 | 24 | 12 | 1027 | 25 | 1 | 28 | 33 | 40 | 922 | 9 | 23 | 41 | 4 | 47 |

**Table 4.6:** Table of the Top 40 Bigrams for the ATT48.
Edge (16,26) is shown in the edge column as | 16 | 26 |. Highlighted cells are global edges.

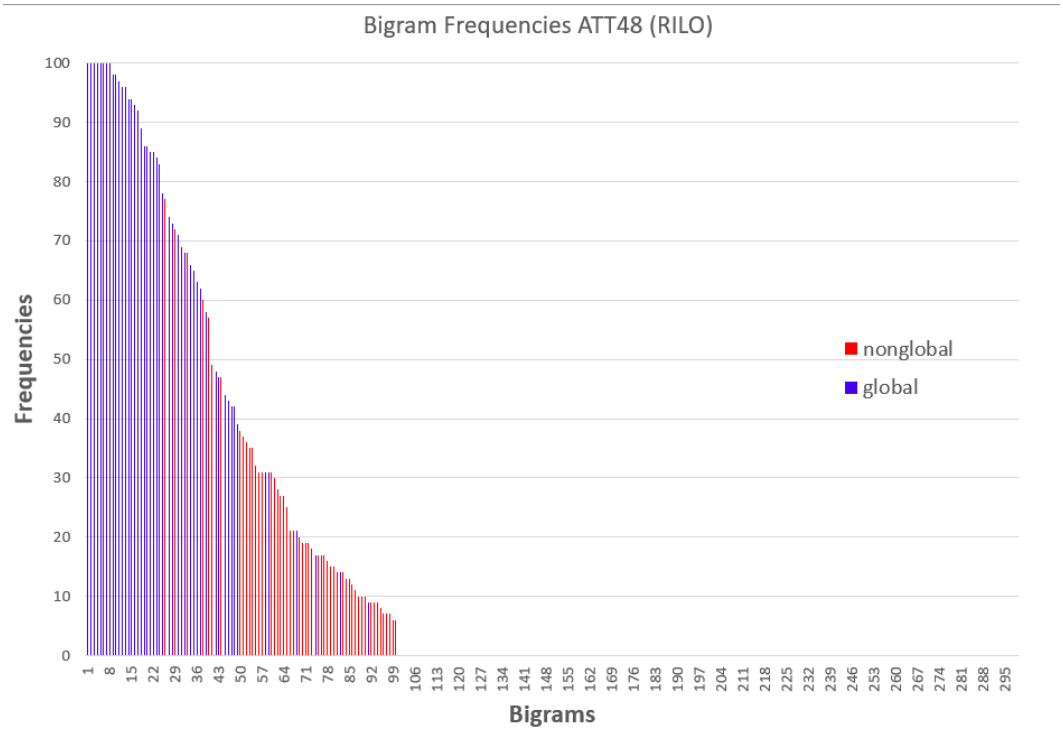
| ATT48 Top 40 Bigrams | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Random | | | NN | | | ACO | | | ACO 2-opt | | | |
| edge | | freq | edge | | freq | edge | | freq | edge | | freq | |
| 16 | 26 | 100 | 16 | 42 | 100 | 31 | 38 | 100 | 5 | 36 | 100 | |
| 16 | 42 | 100 | 18 | 26 | 100 | 9 | 23 | 100 | 18 | 26 | 100 | |
| 18 | 26 | 100 | 18 | 36 | 100 | 34 | 44 | 100 | 18 | 36 | 100 | |
| 18 | 36 | 100 | 5 | 36 | 100 | 3 | 25 | 100 | 16 | 26 | 100 | |
| 3 | 25 | 100 | 3 | 25 | 100 | 0 | 7 | 99 | 16 | 42 | 100 | |
| 3 | 34 | 100 | 34 | 44 | 100 | 23 | 31 | 99 | 9 | 44 | 100 | |
| 34 | 44 | 100 | 16 | 26 | 99 | 9 | 41 | 99 | 34 | 44 | 100 | |
| 5 | 36 | 100 | 15 | 21 | 99 | 41 | 44 | 99 | 3 | 34 | 100 | |
| 0 | 7 | 98 | 4 | 47 | 99 | 25 | 34 | 99 | 3 | 25 | 100 | |
| 15 | 21 | 98 | 0 | 7 | 97 | 1 | 28 | 99 | 1 | 28 | 100 | |
| 30 | 43 | 97 | 3 | 34 | 97 | 4 | 28 | 99 | 17 | 43 | 99 | |
| 29 | 42 | 96 | 31 | 38 | 94 | 4 | 47 | 99 | 6 | 17 | 99 | |
| 5 | 27 | 96 | 30 | 43 | 92 | 15 | 40 | 99 | 31 | 38 | 99 | |
| 1 | 28 | 94 | 29 | 42 | 91 | 1 | 3 | 98 | 33 | 40 | 99 | |
| 31 | 38 | 94 | 5 | 27 | 91 | 33 | 40 | 98 | 0 | 7 | 99 | |
| 33 | 40 | 93 | 30 | 37 | 89 | 16 | 42 | 96 | 15 | 21 | 97 | |
| 4 | 47 | 92 | 32 | 45 | 88 | 32 | 45 | 94 | 5 | 27 | 97 | |
| 30 | 37 | 89 | 33 | 40 | 88 | 15 | 21 | 93 | 4 | 28 | 97 | |
| 17 | 43 | 86 | 13 | 24 | 87 | 18 | 36 | 91 | 29 | 42 | 96 | |
| 6 | 17 | 86 | 1 | 28 | 84 | 6 | 17 | 89 | 30 | 43 | 94 | |
| 29 | 35 | 85 | 6 | 17 | 83 | 7 | 8 | 88 | 15 | 40 | 94 | |
| 6 | 27 | 85 | 17 | 43 | 82 | 17 | 43 | 86 | 30 | 37 | 88 | |
| 13 | 24 | 84 | 2 | 21 | 75 | 8 | 37 | 86 | 19 | 46 | 88 | |
| 9 | 44 | 83 | 9 | 23 | 74 | 11 | 14 | 85 | 8 | 37 | 87 | |
| 32 | 45 | 78 | 9 | 44 | 73 | 19 | 46 | 82 | 32 | 45 | 85 | |
| 7 | 8 | 77 | 10 | 22 | 71 | 0 | 21 | 78 | 13 | 24 | 84 | |
| 19 | 46 | 74 | 12 | 20 | 69 | 13 | 24 | 78 | 11 | 14 | 79 | |
| 2 | 21 | 73 | 11 | 14 | 69 | 29 | 42 | 75 | 4 | 47 | 77 | |
| 10 | 22 | 72 | 29 | 35 | 67 | 33 | 47 | 72 | 7 | 8 | 77 | |
| 14 | 39 | 71 | 6 | 27 | 67 | 2 | 22 | 72 | 6 | 27 | 75 | |
| 9 | 23 | 69 | 7 | 8 | 67 | 30 | 37 | 67 | 29 | 35 | 75 | |
| 20 | 46 | 68 | 0 | 15 | 64 | 18 | 26 | 64 | 23 | 31 | 75 | |
| 8 | 37 | 68 | 38 | 47 | 63 | 12 | 20 | 63 | 10 | 22 | 75 | |
| 19 | 32 | 66 | 19 | 46 | 62 | 2 | 39 | 63 | 12 | 20 | 74 | |
| 11 | 14 | 65 | 13 | 33 | 61 | 30 | 43 | 63 | 14 | 39 | 73 | |
| 35 | 45 | 63 | 20 | 46 | 61 | 19 | 32 | 60 | 35 | 45 | 71 | |
| 1 | 25 | 62 | 8 | 37 | 58 | 14 | 39 | 59 | 9 | 23 | 67 | |
| 12 | 20 | 60 | 12 | 24 | 56 | 30 | 45 | 59 | 2 | 22 | 66 | |
| 10 | 11 | 58 | 19 | 32 | 56 | 6 | 35 | 58 | 19 | 32 | 61 | |
| 23 | 31 | 57 | 10 | 11 | 55 | 10 | 22 | 57 | 10 | 11 | 61 | |

**Table 4.7:** Table of the Top 40 Trigrams for the ATT48.
Edge (3, 34, 44) is shown in the edge column as | 3 | 34 | 44 |. Highlighted cells are global trigrams.

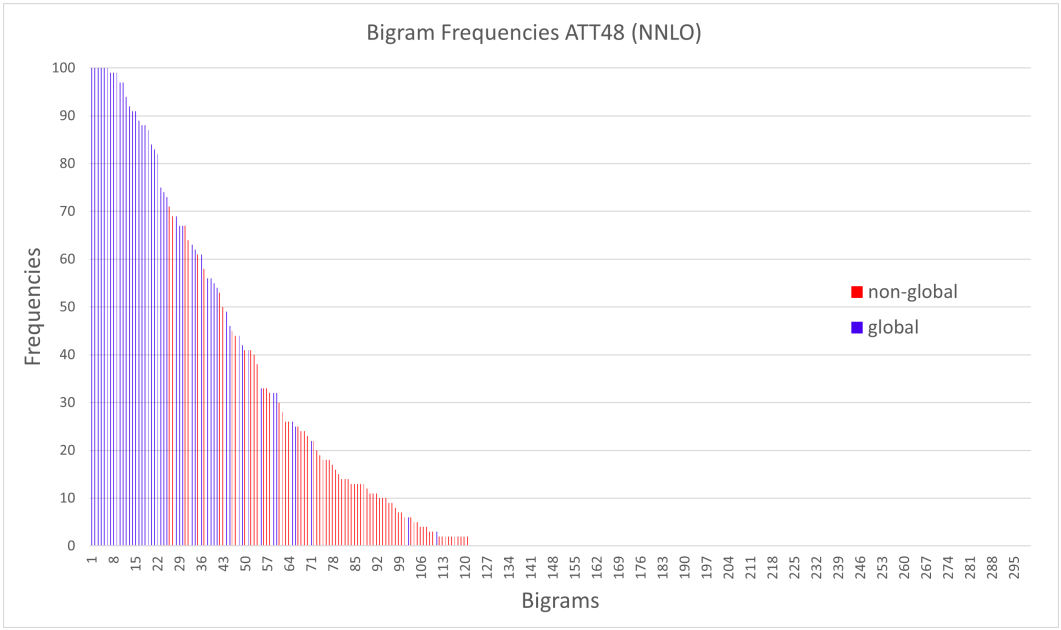
| ATT48 Top 40 Trigrams | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Random | | | | NN | | | | ACO | | | | ACO 2-opt | | | |
| edge | | | freq | edge | | | freq | edge | | | freq | edge | | | freq |
| 3 | 34 | 44 | 99 | 26 | 16 | 42 | 98 | 23 | 31 | 38 | 99 | 9 | 44 | 34 | 100 |
| 5 | 36 | 18 | 99 | 5 | 36 | 18 | 98 | 23 | 9 | 41 | 99 | 3 | 34 | 44 | 100 |
| 26 | 18 | 36 | 98 | 16 | 26 | 18 | 97 | 34 | 44 | 41 | 99 | 25 | 3 | 34 | 100 |
| 26 | 16 | 42 | 97 | 25 | 3 | 34 | 96 | 25 | 34 | 44 | 99 | 6 | 17 | 43 | 97 |
| 25 | 3 | 34 | 96 | 3 | 34 | 44 | 95 | 9 | 23 | 31 | 98 | 5 | 36 | 18 | 97 |
| 16 | 26 | 18 | 96 | 26 | 18 | 36 | 92 | 9 | 41 | 44 | 98 | 26 | 18 | 36 | 97 |
| 16 | 42 | 29 | 95 | 16 | 42 | 29 | 91 | 3 | 25 | 34 | 98 | 26 | 16 | 42 | 97 |
| 27 | 5 | 36 | 94 | 27 | 5 | 36 | 91 | 1 | 3 | 25 | 98 | 16 | 42 | 29 | 96 |
| 6 | 17 | 43 | 86 | 6 | 17 | 43 | 81 | 3 | 1 | 28 | 98 | 1 | 28 | 4 | 96 |
| 35 | 29 | 42 | 85 | 37 | 30 | 43 | 78 | 28 | 4 | 47 | 98 | 16 | 26 | 18 | 94 |
| 37 | 30 | 43 | 85 | 2 | 21 | 15 | 73 | 15 | 40 | 33 | 97 | 17 | 43 | 30 | 93 |
| 5 | 27 | 6 | 83 | 17 | 43 | 30 | 73 | 1 | 28 | 4 | 96 | 15 | 40 | 33 | 93 |
| 9 | 44 | 34 | 82 | 9 | 44 | 34 | 73 | 21 | 15 | 40 | 92 | 21 | 15 | 40 | 91 |
| 17 | 43 | 30 | 80 | 23 | 9 | 44 | 71 | 0 | 7 | 8 | 87 | 27 | 5 | 36 | 90 |
| 0 | 7 | 8 | 76 | 35 | 29 | 42 | 67 | 7 | 8 | 37 | 81 | 37 | 30 | 43 | 81 |
| 2 | 21 | 15 | 70 | 5 | 27 | 6 | 66 | 7 | 0 | 21 | 77 | 0 | 7 | 8 | 77 |
| 17 | 6 | 27 | 69 | 0 | 7 | 8 | 65 | 6 | 17 | 43 | 75 | 8 | 37 | 30 | 76 |
| 23 | 9 | 44 | 68 | 4 | 47 | 38 | 63 | 16 | 42 | 29 | 74 | 5 | 27 | 6 | 74 |
| 1 | 25 | 3 | 62 | 7 | 0 | 15 | 61 | 0 | 21 | 15 | 72 | 35 | 29 | 42 | 74 |
| 29 | 35 | 45 | 62 | 0 | 15 | 21 | 61 | 4 | 47 | 33 | 71 | 23 | 31 | 38 | 74 |
| 25 | 1 | 28 | 56 | 31 | 38 | 47 | 57 | 40 | 33 | 47 | 71 | 28 | 4 | 47 | 74 |
| 8 | 37 | 30 | 56 | 7 | 8 | 37 | 50 | 8 | 37 | 30 | 59 | 7 | 8 | 37 | 74 |
| 7 | 8 | 37 | 55 | 13 | 33 | 40 | 49 | 26 | 18 | 36 | 57 | 17 | 6 | 27 | 73 |
| 32 | 45 | 35 | 54 | 17 | 6 | 27 | 49 | 30 | 45 | 32 | 56 | 29 | 35 | 45 | 68 |
| 23 | 31 | 38 | 51 | 8 | 37 | 30 | 48 | 19 | 32 | 45 | 54 | 23 | 9 | 44 | 67 |
| 0 | 15 | 21 | 48 | 24 | 13 | 33 | 47 | 17 | 43 | 30 | 52 | 25 | 1 | 28 | 57 |
| 7 | 0 | 15 | 47 | 23 | 31 | 38 | 44 | 32 | 19 | 46 | 50 | 1 | 25 | 3 | 56 |
| 19 | 46 | 20 | 46 | 20 | 31 | 38 | 43 | 20 | 38 | 31 | 50 | 32 | 19 | 46 | 52 |
| 21 | 15 | 40 | 45 | 19 | 32 | 45 | 43 | 17 | 6 | 35 | 49 | 11 | 14 | 39 | 52 |
| 19 | 32 | 45 | 45 | 29 | 35 | 45 | 42 | 26 | 16 | 42 | 49 | 7 | 0 | 21 | 52 |
| 32 | 19 | 46 | 42 | 1 | 25 | 3 | 42 | 11 | 14 | 39 | 46 | 32 | 45 | 35 | 51 |
| 1 | 28 | 4 | 41 | 12 | 24 | 13 | 41 | 22 | 2 | 39 | 46 | 0 | 21 | 15 | 50 |
| 11 | 14 | 39 | 40 | 28 | 1 | 41 | 41 | 24 | 38 | 31 | 45 | 10 | 11 | 14 | 46 |
| 15 | 40 | 33 | 39 | 28 | 4 | 47 | 40 | 22 | 13 | 24 | 39 | 19 | 32 | 45 | 45 |
| 31 | 38 | 47 | 38 | 32 | 45 | 35 | 37 | 2 | 22 | 10 | 36 | 4 | 47 | 33 | 45 |
| 28 | 4 | 47 | 37 | 28 | 40 | 33 | 35 | 5 | 36 | 18 | 36 | 40 | 33 | 47 | 45 |
| 9 | 23 | 41 | 36 | 9 | 23 | 31 | 35 | 10 | 11 | 14 | 35 | 2 | 22 | 10 | 45 |
| 4 | 47 | 38 | 35 | 25 | 1 | 28 | 34 | 36 | 16 | 42 | 34 | 2 | 21 | 15 | 44 |
| 10 | 11 | 14 | 33 | 1 | 28 | 40 | 33 | 43 | 30 | 45 | 32 | 19 | 46 | 20 | 43 |
| 20 | 31 | 38 | 33 | 6 | 35 | 27 | 33 | 16 | 36 | 18 | 31 | 28 | 1 | 41 | 43 |

**Table 4.8:** Table of the Top 40 5-grams for the ATT48. Highlighted cells are global edges.

| ATT48 Top 40 5-grams | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Random | | | | | | | Nearest Neighbors | | | | | | ACO 2-opt | | | | | |
| edge | | | | | | freq | edge | | | | | freq | edge | | | | | freq |
| 5 | 36 | 18 | 26 | 16 | 93 | | 18 | 26 | 16 | 42 | 29 | 88 | 9 | 44 | 34 | 3 | 25 | 100 |
| 26 | 18 | 36 | 5 | 27 | 91 | | 36 | 18 | 26 | 16 | 42 | 88 | 5 | 36 | 18 | 26 | 16 | 87 |
| 36 | 18 | 26 | 16 | 42 | 91 | | 5 | 36 | 18 | 26 | 16 | 87 | 36 | 18 | 26 | 16 | 42 | 87 |
| 18 | 26 | 16 | 42 | 29 | 88 | | 26 | 18 | 36 | 5 | 27 | 81 | 18 | 26 | 16 | 42 | 29 | 87 |
| 26 | 16 | 42 | 29 | 35 | 81 | | 3 | 34 | 44 | 9 | 23 | 70 | 26 | 18 | 36 | 5 | 27 | 84 |
| 6 | 27 | 5 | 36 | 18 | 80 | | 9 | 44 | 34 | 3 | 25 | 70 | 6 | 17 | 43 | 30 | 37 | 78 |
| 9 | 44 | 34 | 3 | 25 | 77 | | 26 | 16 | 42 | 29 | 35 | 67 | 26 | 16 | 42 | 29 | 35 | 73 |
| 6 | 17 | 43 | 30 | 37 | 70 | | 6 | 27 | 5 | 36 | 18 | 65 | 0 | 7 | 8 | 37 | 30 | 73 |
| 5 | 27 | 6 | 17 | 43 | 67 | | 6 | 17 | 43 | 30 | 37 | 58 | 27 | 6 | 17 | 43 | 30 | 71 |
| 3 | 34 | 44 | 9 | 23 | 66 | | 2 | 21 | 15 | 0 | 7 | 56 | 5 | 27 | 6 | 17 | 43 | 70 |
| 27 | 6 | 17 | 43 | 30 | 66 | | 0 | 7 | 8 | 37 | 30 | 48 | 8 | 37 | 30 | 43 | 17 | 70 |
| 17 | 6 | 27 | 5 | 36 | 65 | | 27 | 6 | 17 | 43 | 30 | 48 | 6 | 27 | 5 | 36 | 18 | 68 |
| 16 | 42 | 29 | 35 | 45 | 61 | | 5 | 27 | 6 | 17 | 43 | 48 | 7 | 8 | 37 | 30 | 43 | 68 |
| 1 | 25 | 3 | 34 | 44 | 60 | | 17 | 6 | 27 | 5 | 36 | 48 | 17 | 6 | 27 | 5 | 36 | 67 |
| 28 | 1 | 25 | 3 | 34 | 54 | | 7 | 8 | 37 | 30 | 43 | 47 | 16 | 42 | 29 | 35 | 45 | 67 |
| 32 | 45 | 35 | 29 | 42 | 53 | | 8 | 7 | 0 | 15 | 21 | 47 | 3 | 34 | 44 | 9 | 23 | 67 |
| 0 | 7 | 8 | 37 | 30 | 52 | | 4 | 47 | 38 | 31 | 20 | 43 | 1 | 25 | 3 | 34 | 44 | 56 |
| 7 | 8 | 37 | 30 | 43 | 52 | | 16 | 42 | 29 | 35 | 45 | 42 | 28 | 1 | 25 | 3 | 34 | 56 |
| 8 | 37 | 30 | 43 | 17 | 46 | | 1 | 25 | 3 | 34 | 44 | 40 | 3 | 25 | 1 | 28 | 4 | 53 |
| 2 | 21 | 15 | 0 | 7 | 43 | | 15 | 0 | 7 | 8 | 37 | 39 | 32 | 45 | 35 | 29 | 42 | 49 |
| 19 | 32 | 45 | 35 | 29 | 37 | | 8 | 37 | 30 | 43 | 17 | 36 | 0 | 21 | 15 | 40 | 33 | 49 |
| 8 | 7 | 0 | 15 | 21 | 36 | | 32 | 45 | 35 | 29 | 42 | 35 | 7 | 0 | 21 | 15 | 40 | 49 |
| 34 | 44 | 9 | 23 | 41 | 34 | | 38 | 31 | 23 | 9 | 44 | 34 | 1 | 28 | 4 | 47 | 33 | 45 |
| 31 | 23 | 9 | 44 | 34 | 28 | | 31 | 23 | 9 | 44 | 34 | 34 | 28 | 4 | 47 | 33 | 40 | 45 |
| 15 | 0 | 7 | 8 | 37 | 27 | | 28 | 1 | 25 | 3 | 34 | 32 | 3 | 25 | 41 | 1 | 28 | 43 |
| 35 | 45 | 32 | 19 | 46 | 26 | | 27 | 35 | 6 | 17 | 43 | 30 | 1 | 41 | 25 | 3 | 34 | 43 |
| 4 | 41 | 23 | 9 | 44 | 24 | | 41 | 25 | 3 | 34 | 44 | 30 | 41 | 25 | 3 | 34 | 44 | 43 |
| 9 | 23 | 41 | 4 | 47 | 24 | | 34 | 44 | 9 | 23 | 41 | 29 | 4 | 47 | 33 | 40 | 15 | 43 |
| 38 | 31 | 23 | 9 | 44 | 24 | | 1 | 41 | 25 | 3 | 34 | 27 | 21 | 15 | 40 | 33 | 47 | 43 |
| 7 | 0 | 21 | 15 | 40 | 23 | | 19 | 32 | 45 | 35 | 29 | 26 | 19 | 32 | 45 | 35 | 29 | 42 |
| 4 | 47 | 38 | 31 | 20 | 22 | | 34 | 3 | 25 | 9 | 41 | 25 | 41 | 1 | 28 | 4 | 47 | 42 |
| 20 | 46 | 19 | 32 | 45 | 21 | | 9 | 25 | 3 | 34 | 44 | 24 | 4 | 28 | 1 | 41 | 25 | 42 |
| 3 | 25 | 1 | 28 | 40 | 20 | | 15 | 21 | 2 | 40 | 33 | 24 | 31 | 23 | 9 | 44 | 34 | 42 |
| 8 | 7 | 0 | 21 | 15 | 20 | | 0 | 15 | 21 | 2 | 40 | 24 | 38 | 31 | 23 | 9 | 44 | 42 |
| 0 | 21 | 15 | 40 | 33 | 19 | | 6 | 35 | 27 | 5 | 36 | 24 | 2 | 21 | 15 | 40 | 33 | 41 |
| 2 | 21 | 15 | 40 | 33 | 18 | | 9 | 23 | 41 | 4 | 47 | 24 | 8 | 7 | 0 | 21 | 15 | 39 |
| 3 | 25 | 1 | 28 | 4 | 18 | | 23 | 44 | 34 | 3 | 25 | 23 | 21 | 0 | 7 | 8 | 37 | 36 |
| 15 | 21 | 2 | 33 | 40 | 18 | | 5 | 27 | 35 | 6 | 17 | 23 | 35 | 45 | 32 | 19 | 46 | 35 |
| 41 | 25 | 3 | 34 | 44 | 18 | | 18 | 36 | 5 | 27 | 35 | 23 | 23 | 41 | 9 | 44 | 34 | 33 |
| 0 | 15 | 21 | 2 | 33 | 17 | | 19 | 29 | 42 | 16 | 26 | 23 | 3 | 34 | 44 | 9 | 41 | 33 |

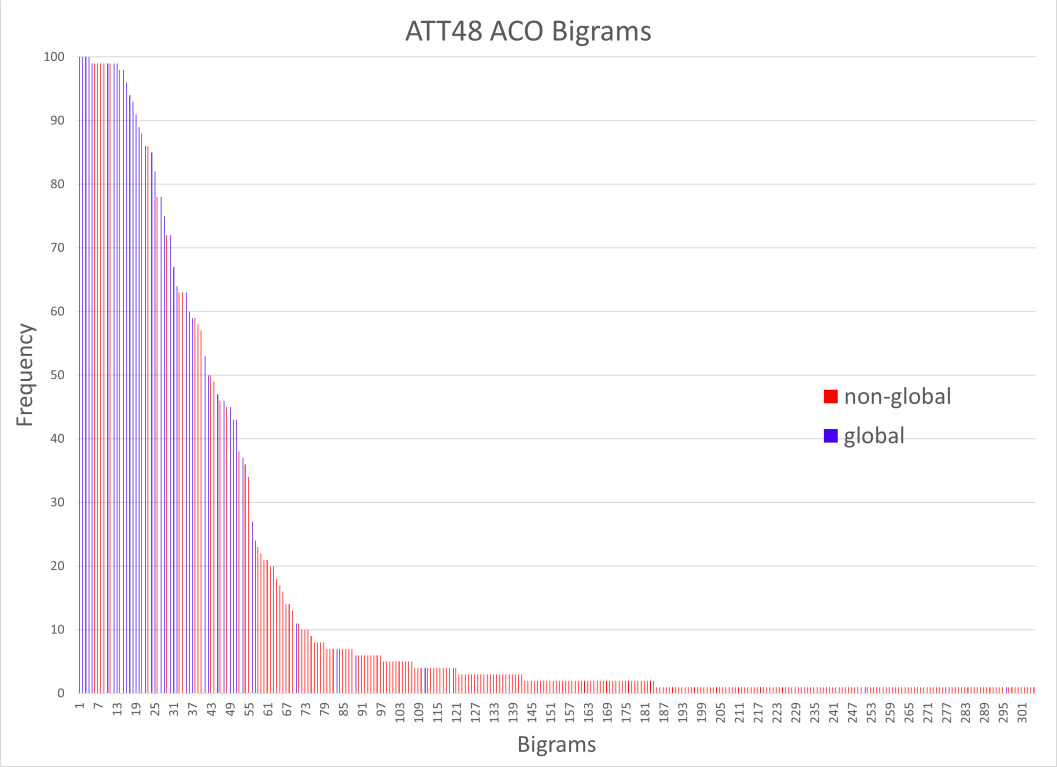



**(a)** ATT48 Bigrams RILO

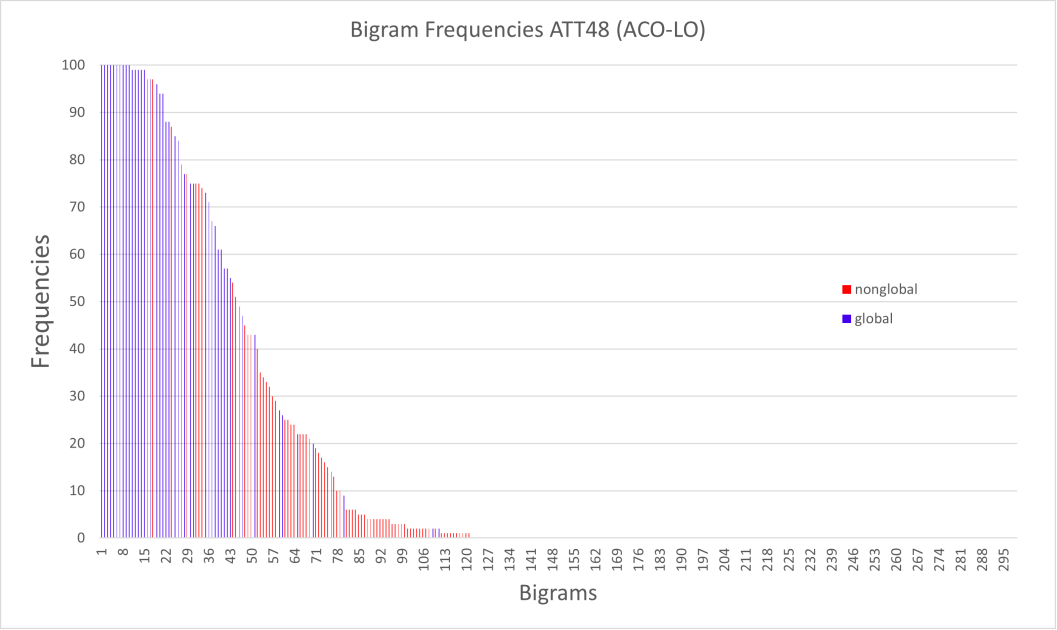



**(b)** ATT48 Bigrams NNLO

**Figure 4.1:** Bigram Frequencies for the ATT48 vs. the Number of Bigrams (a) RILO (b) NNLO Note: both sub-figures are truncated at ten after the last globally optimal bigram.

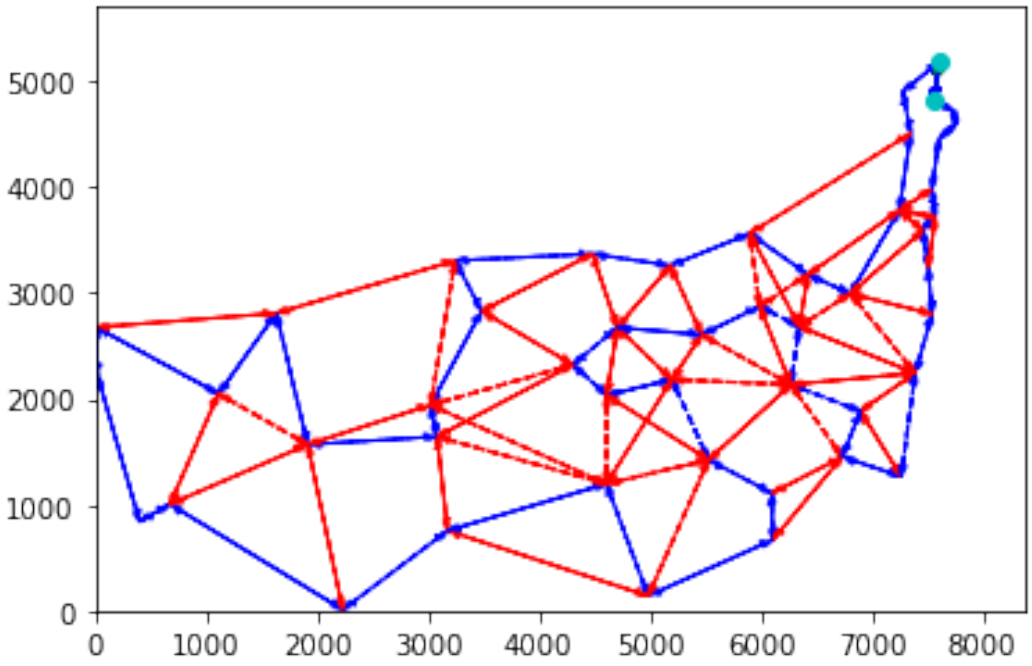



**(a)** ATT48 Bigrams ACO-S1

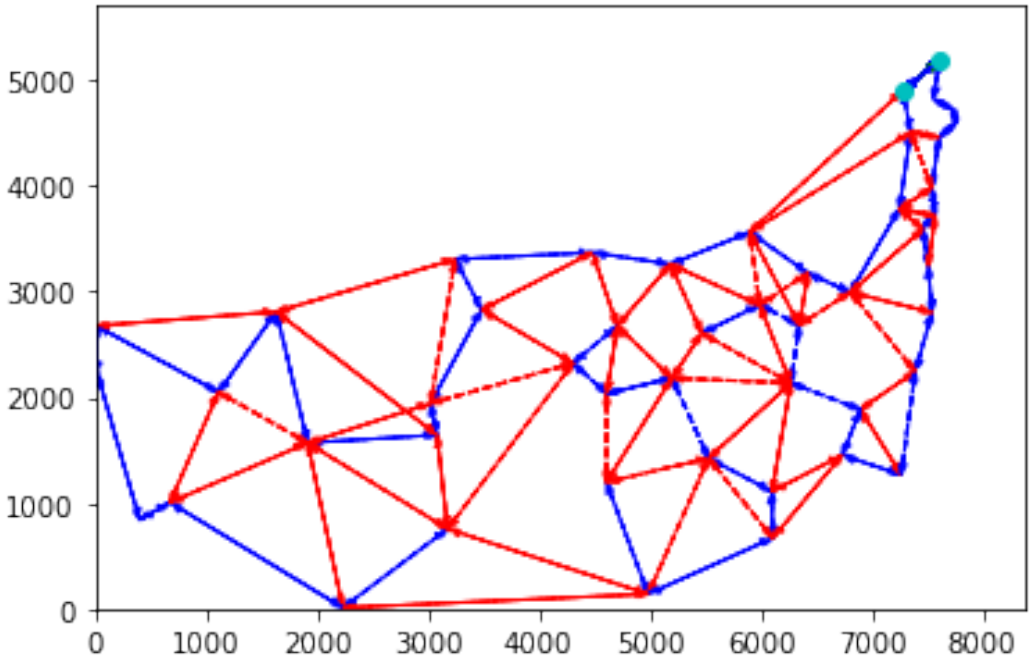



**(b)** ATT48 Bigrams ACO-LO 2-opt

**Figure 4.2:** Bigram Frequencies for the ATT48 vs. the Number of Bigrams (a) ACO-S1 (b) ACO-LO Note: Both sub-figures are truncated at ten after the last globally optimal bigram.

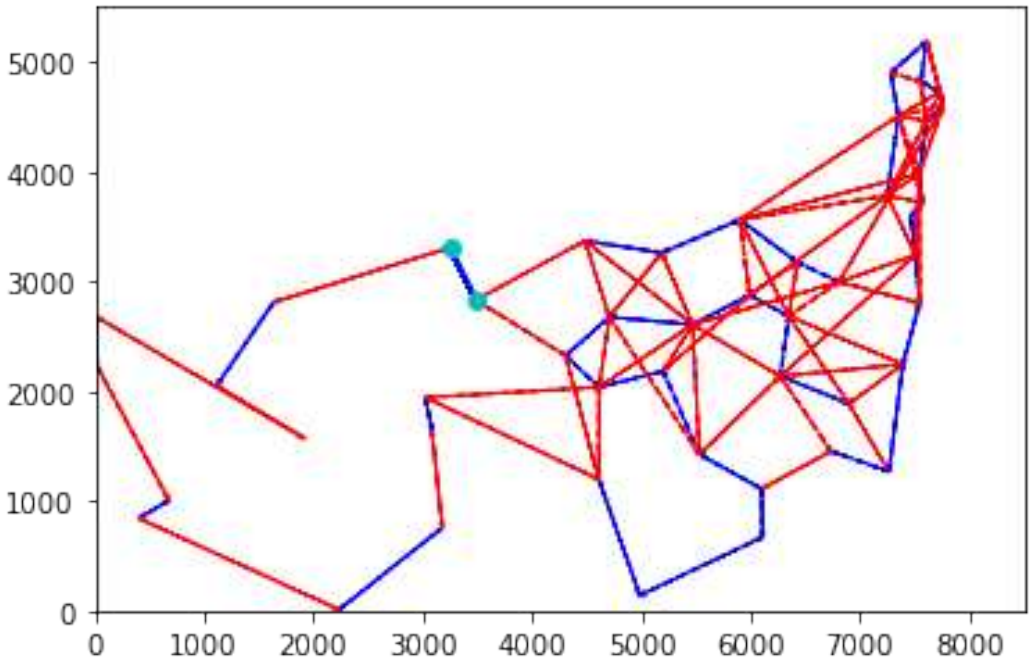



**(a)** ATT48 Top 100 Bigrams RILO

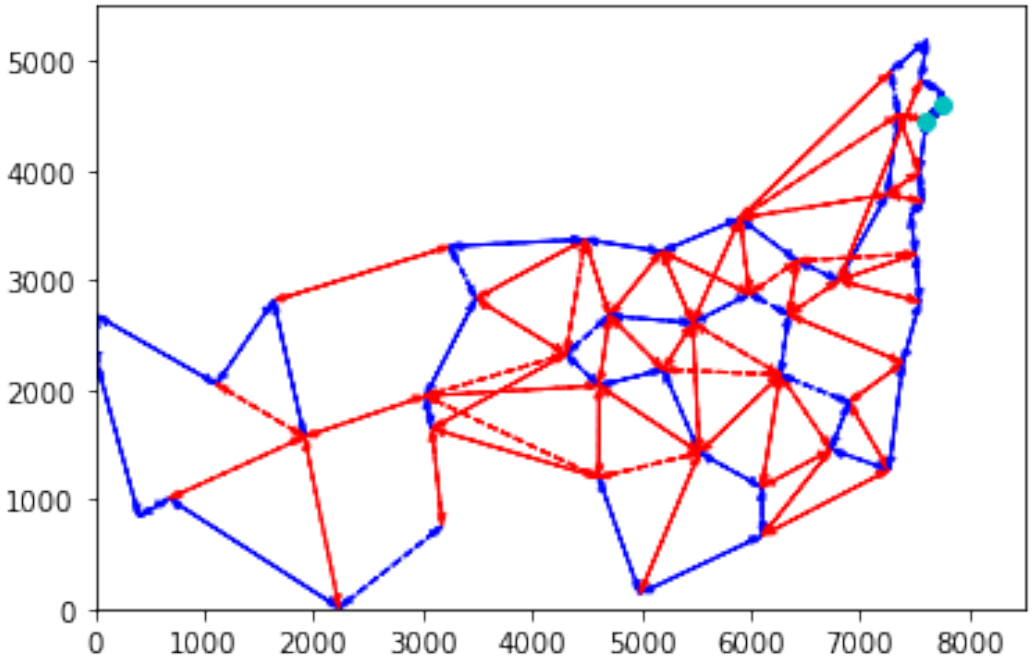



**(b)** ATT48 Top 100 Bigrams NNLO

**Figure 4.3:** Top 100 ATT48 Bigrams (a) RILO (b) NNLO
Note: blue edges are globally optimal edges while red are locally optimal edges. The edge with teal vertices is the first ranked edge.

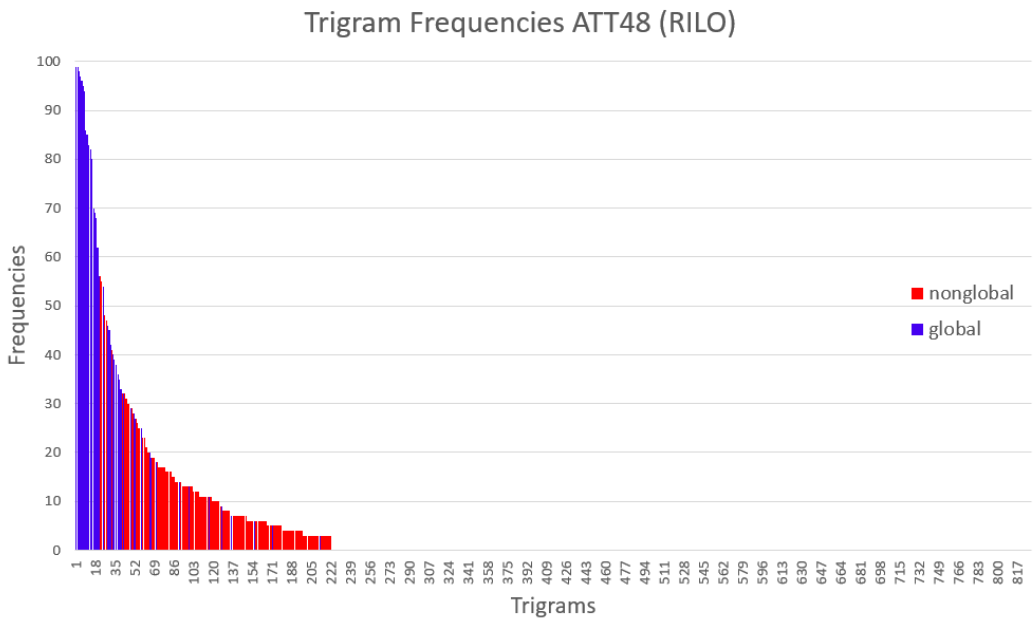



**(a)** ATT48 Top 100 Bigrams ACO-S1

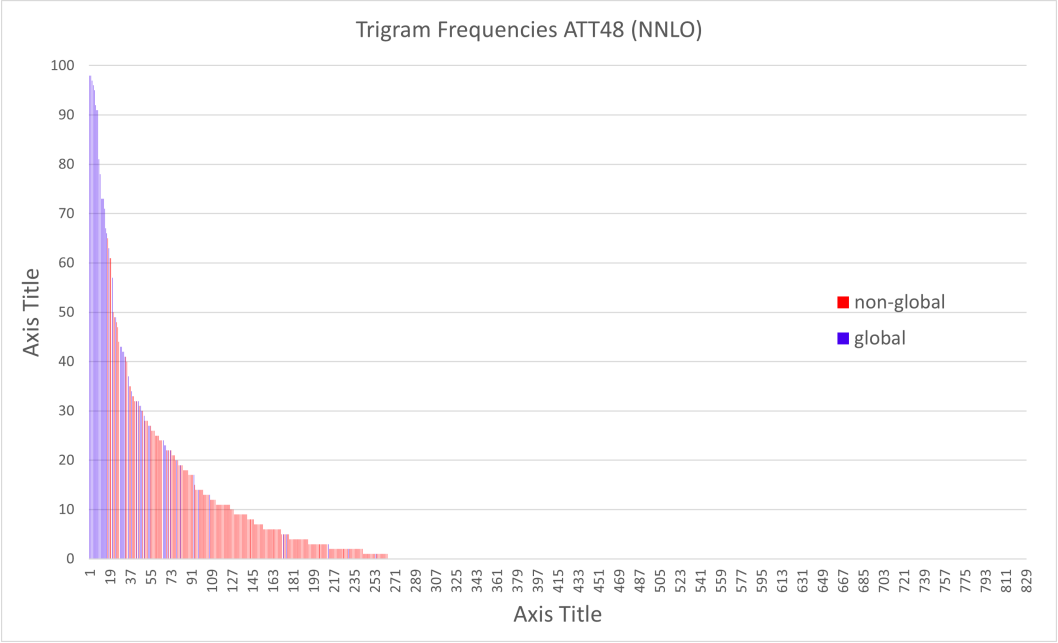



**(b)** ATT48 Top 100 Bigrams ACO-LO

**Figure 4.4:** Top 100 ATT48 Bigrams (a) ACO-S1 (b) ACO-LO
Note: blue edges are globally optimal edges while red are locally optimal edges. The edge with teal vertices is the first ranked edge.

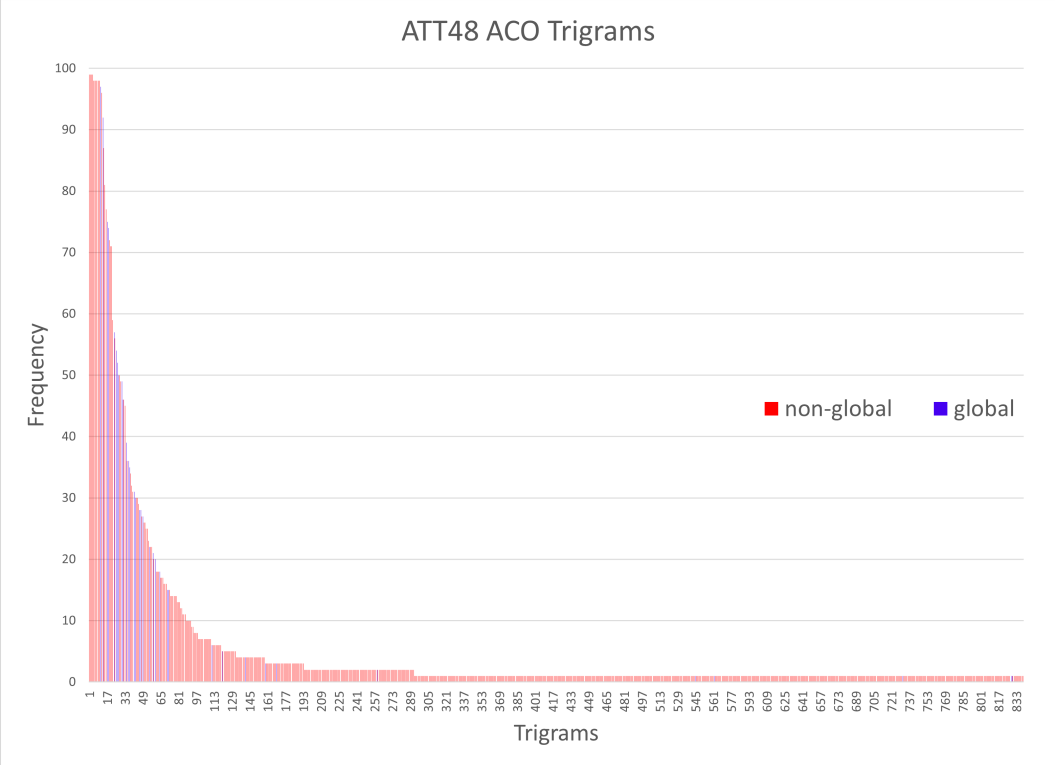



**(a)** ATT48 Trigrams RILO

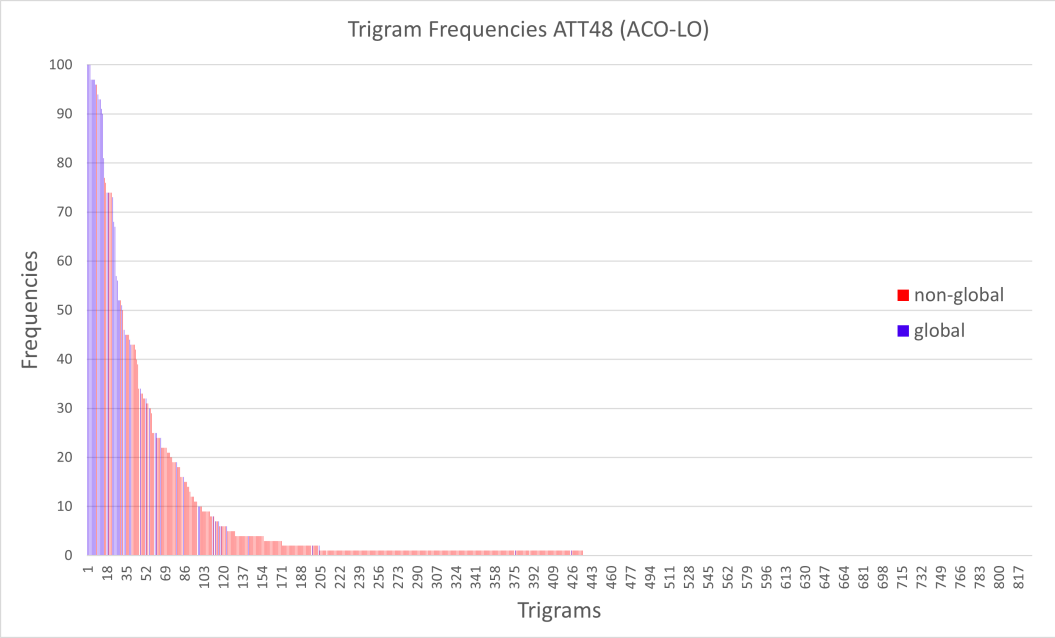



**(b)** ATT48 Trigrams NNLO

**Figure 4.5:** Trigram Frequencies for the ATT48 vs. the Number of Trigrams (a) RILO (b) NNLO Note: both figures are truncated ten after the last globally optimal trigram.




**(a)** ATT48 Trigrams ACO-S1




**(b)** ATT48 Trigrams ACO-LO

**Figure 4.6:** Trigram Frequencies for the ATT48 vs. the Number of Trigrams (a) ACO-S1 (b) ACO-LO

## 4.2 EIL101 Frequencies

Similar to the ATT48, the EIL101 is another city problem. We wanted to examine different city problems to see if we achieved similar results with problems of different sizes. Instead of 48 cities, the EIL101 has 101 cities in its set. N-grams from local optima generated with the different tour construction methods were examined for their frequencies and percentages for the EIL101. The same procedure of sorting and color coding the results in Excel was followed so that we could analyze and plot the results. The n-grams were sorted from highest to lowest frequency and given ranks based on this sorting. Again, the tables include the number of global edges found by each set of local optima, the percentage of global frequencies, and sorted n-grams of different sizes. The highest frequency edge corresponds with rank 1 in the tables of sorted n-grams.

**Table 4.9:** Table of the Number of Global Edges Found for Each Type of Tour Construction

| **EIL101 Global Edges Count** | | | | |
|---|---|---|---|---|
| **N-grams** | **RILO** | **NNLO** | **ACO-S1** | **ACO-LO** |
| 2 | 101 | 101 | 98 | 101 |
| 3 | 100 | 101 | 91 | 101 |
| 4 | 99 | 101 | 75 | 99 |
| 5 | 97 | 96 | 56 | 96 |
| 7 | 90 | 84 | 26 | 83 |

**Table 4.10:** Table of the Percent of Global Edges for Each Tour for the EIL101.

| **EIL101 Percentage of Global N-Grams** | | | | |
|---|---|---|---|---|
| **N-grams** | **RILO** | **NNLO** | **ACO-S1** | **ACO-LO** |
| 2 | 61.41% | 59.39% | 51.23% | 60.33% |
| 3 | 32.88% | 30.32% | 21.21% | 31.63% |
| 4 | 19.33% | 17.39% | 9.65% | 17.94% |
| 5 | 11.63% | 10.10% | 4.92% | 11.24% |
| 7 | 4.98% | 4.38% | 1.07% | 4.75% |

In Table 4.9, the number of global edges found within each set of 100 local optima are shown. For the EIL101, the total number of global edges is 101, and the NNLO found all the global edges for the bigrams, trigrams, and 4-grams. The RILO contained all of the bigrams, but missed one trigram and two 4-grams. However, the RILO found the most 5-grams and 7-grams. ACO-LO gathered all of the bigrams and trigrams and found the same number of 4-grams as RILO (99), but this set found one less 5-gram (96) and only found 83 in the 7-grams. RILO found 90 in the 7-grams.

In Table 4.10, the percentage of global frequencies out of the total frequencies for each n-gram size are shown. RILO and ACO-LO had the highest and second highest percentages for each size of n-gram. However, NNLO was the only method that collected 101 of the global edges for n-grams of size 2 through 4 in Table 4.9. These two tables are interesting to compare to see how successful each type of tour construction method was at collecting global edges.

In Table 4.11, the top 20 and bottom 20 ranked global bigrams for each type of tour show how far you have to traverse the edge tables to produce all of the global edges. RILO produced all of the global edges by the 168th frequency, followed by ACO-LO at 185, NNLO at 188, and lastly ACO-S1 at 941. These results make sense when we consider that RILO are unbiased while NNLO and ACO both have a bias in the way they construct tours. In Table 4.12, the top 20 and bottom 20 global trigrams are featured. Once again, RILO collected all of the global trigrams first by the 540th frequency, followed by ACO-LO by the 608th, NNLO by the 698th, and ACO-S1 by the 2602nd. With the 5-grams, in Table 4.13, RILO had the last global 5-gram at the 3960th frequency, NNLO at the 3171st, and ACO-LO at 3057th.

When looking at the tables of n-grams, it is interesting to note the frequencies of global edges and where they appear in the top 40. For example, in Table 4.14, RILO and ACO-LO both have 3 non-global edges in the top 40, NNLO have 5, and ACO-S1 has 9. The trigrams are shown in Table 4.15. While all but ACO-S1 have mostly global edges in the top 40, RILO have the largest group of global edges at the top as well as the fewest number of non-global edges in the top 40. When examining the 5-grams in Table 4.16, we start to get less consistency with the representa-

tion of global edges in the top 40 for all three: RILO, NNLO, and ACO-LO. ACO without 2-opt is in the Appendix.

After the tables, the frequency of n-grams and the number of n-grams plots highlight the spread of global and non-global edges in each frequency table. Red lines are non-global edges while blue are global edges. To be able to see the details better, each plot was stopped 10 n-grams after the last found global edge. In Figure 4.7, the bigram frequencies for RRLO and NNLO are shown, and the last global edge for RILO is at 168 while the last for NNLO is at 188. In Figure 4.8, ACO-S1 and ACO-LO are shown. ACO-S1 has a much larger spread of the global edges and collects the last global bigram at 941 while ACO-LO have the last one by 185.

In the graphs in Figures 4.9 and 4.10, the global edges (blue) and the non-global edges (red) from the top 200 bigrams for RRLO, NNLO, ACO-S1, and ACO-LO are shown. The edge with the teal vertices in each graph is the first ranked bigram, and it is interesting that it is the same edge in all four graphs. After that, we can see some differences in how much of the global optimum is represented as well as which edges are in the top 200 of each sample.

When examining the trigrams, in Figures 4.11 and 4.12, we see that RRLO have the last global trigram at 540, NNLO at 698, ACO-S1 at 2602, and ACO-LO at 608. Again, RILO collected all the global trigrams the earliest.

To summarize the results for the EIL101, RILO does not always collect the most total global edges, but this sample does collect the global n-grams at higher rates than the other methods based on percentages and it collects them at earlier high frequencies than the other methods in most cases. The bigrams and trigrams are collected earlier in the frequency table than the other methods, but the 5-grams are sampled later for RILO than NNLO and ACO-LO.

**Table 4.11:** Table of Top 20 and Bottom 20 Global Bigrams for the EIL101
Edge (24,54) is shown in the edge column as | 24 | 54 |.

| **EIL101 Top 20, Bottom 20 Global Bigrams** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **RILO** | | | **NNLO** | | | **ACO-S1** | | | **ACO-LO** | | |
| rank | edge | | rank | edge | | rank | edge | | rank | edge | |
| 1 | 24 | 54 | 1 | 24 | 54 | 1 | 24 | 54 | 1 | 24 | 54 |
| 2 | 31 | 89 | 2 | 33 | 77 | 2 | 36 | 97 | 2 | 14 | 42 |
| 3 | 33 | 77 | 3 | 22 | 66 | 3 | 33 | 77 | 3 | 33 | 77 |
| 4 | 14 | 42 | 4 | 23 | 28 | 4 | 21 | 73 | 4 | 23 | 28 |
| 5 | 23 | 28 | 5 | 14 | 42 | 5 | 23 | 28 | 5 | 62 | 89 |
| 6 | 21 | 40 | 6 | 31 | 89 | 6 | 14 | 56 | 6 | 35 | 48 |
| 7 | 35 | 48 | 7 | 0 | 68 | 7 | 37 | 43 | 7 | 31 | 89 |
| 8 | 67 | 79 | 8 | 21 | 40 | 8 | 67 | 79 | 8 | 21 | 40 |
| 10 | 22 | 66 | 9 | 67 | 79 | 9 | 48 | 63 | 9 | 38 | 66 |
| 11 | 62 | 89 | 10 | 35 | 48 | 10 | 21 | 40 | 10 | 0 | 68 |
| 12 | 36 | 97 | 11 | 1 | 56 | 13 | 14 | 42 | 11 | 67 | 79 |
| 13 | 20 | 72 | 12 | 15 | 60 | 15 | 1 | 56 | 12 | 48 | 63 |
| 14 | 0 | 68 | 13 | 95 | 98 | 17 | 55 | 74 | 14 | 95 | 98 |
| 15 | 39 | 57 | 15 | 36 | 97 | 19 | 39 | 57 | 15 | 39 | 57 |
| 16 | 15 | 85 | 16 | 5 | 88 | 20 | 59 | 82 | 16 | 29 | 69 |
| 17 | 30 | 87 | 17 | 38 | 66 | 21 | 0 | 68 | 17 | 20 | 72 |
| 18 | 38 | 66 | 18 | 16 | 83 | 22 | 38 | 66 | 18 | 14 | 56 |
| 19 | 52 | 100 | 19 | 30 | 87 | 24 | 31 | 89 | 19 | 64 | 70 |
| 20 | 16 | 83 | 20 | 90 | 99 | 25 | 64 | 70 | 21 | 30 | 87 |
| 21 | 95 | 98 | 21 | 37 | 85 | 26 | 33 | 34 | 22 | 52 | 100 |
| | | | | | | | | | | | |
| 107 | 93 | 94 | 118 | 23 | 53 | 147 | 12 | 93 | 123 | 65 | 70 |
| 110 | 28 | 67 | 120 | 3 | 24 | 153 | 5 | 95 | 125 | 32 | 78 |
| 111 | 36 | 91 | 121 | 10 | 61 | 155 | 32 | 78 | 128 | 20 | 39 |
| 114 | 12 | 57 | 132 | 92 | 98 | 156 | 6 | 81 | 129 | 36 | 91 |
| 117 | 10 | 61 | 134 | 27 | 52 | 158 | 62 | 63 | 130 | 51 | 88 |
| 120 | 32 | 78 | 135 | 32 | 50 | 161 | 25 | 27 | 133 | 97 | 99 |
| 121 | 58 | 92 | 141 | 58 | 92 | 166 | 86 | 96 | 134 | 27 | 52 |
| 122 | 9 | 31 | 144 | 51 | 88 | 167 | 20 | 39 | 135 | 32 | 50 |
| 128 | 20 | 39 | 146 | 13 | 41 | 168 | 53 | 54 | 136 | 37 | 85 |
| 129 | 3 | 38 | 149 | 9 | 31 | 174 | 37 | 85 | 142 | 10 | 61 |
| 135 | 45 | 46 | 153 | 65 | 70 | 176 | 51 | 88 | 144 | 3 | 38 |
| 138 | 97 | 99 | 157 | 40 | 74 | 181 | 92 | 98 | 145 | 13 | 43 |
| 141 | 18 | 47 | 158 | 18 | 47 | 185 | 23 | 53 | 146 | 1 | 86 |
| 142 | 32 | 50 | 162 | 1 | 86 | 190 | 11 | 79 | 149 | 3 | 24 |
| 143 | 1 | 86 | 167 | 3 | 38 | 223 | 13 | 41 | 153 | 18 | 47 |
| 145 | 77 | 80 | 173 | 28 | 67 | 230 | 45 | 46 | 154 | 62 | 63 |
| 151 | 13 | 41 | 174 | 34 | 64 | 234 | 77 | 80 | 156 | 9 | 31 |
| 153 | 3 | 24 | 175 | 5 | 95 | 243 | 13 | 43 | 162 | 58 | 92 |
| 160 | 27 | 52 | 178 | 45 | 46 | 305 | 3 | 38 | 176 | 77 | 80 |
| 168 | 40 | 74 | 188 | 77 | 80 | 941 | 9 | 31 | 185 | 13 | 41 |

**Table 4.12:** Table of Top 20, Bottom 20 Global Trigrams for the EIL101
Edge (31, 89, 62) is shown in the edge column as | 31 | 89 | 62 |.

| **EIL101 Top 20, Bottom 20 Global Trigrams** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RILO** | | | | **NNLO** | | | | **ACO-S1** | | | | **ACO-LO** | | | |
| rank | edge | | | rank | edge | | | rank | edge | | | rank | edge | | |
| 1 | 31 | 89 | 62 | 1 | 22 | 66 | 38 | 1 | 40 | 21 | 73 | 1 | 31 | 89 | 62 |
| 2 | 22 | 66 | 38 | 2 | 60 | 15 | 85 | 3 | 1 | 56 | 14 | 2 | 42 | 14 | 56 |
| 3 | 34 | 33 | 77 | 3 | 15 | 85 | 37 | 5 | 42 | 14 | 56 | 3 | 35 | 48 | 63 |
| 4 | 35 | 48 | 63 | 5 | 31 | 89 | 62 | 6 | 34 | 33 | 77 | 4 | 34 | 33 | 77 |
| 5 | 76 | 2 | 78 | 6 | 49 | 0 | 68 | 9 | 21 | 73 | 71 | 5 | 40 | 21 | 73 |
| 6 | 2 | 76 | 75 | 7 | 34 | 33 | 77 | 10 | 35 | 48 | 63 | 6 | 22 | 66 | 38 |
| 7 | 55 | 22 | 66 | 8 | 4 | 83 | 16 | 12 | 0 | 68 | 26 | 7 | 1 | 56 | 14 |
| 8 | 60 | 15 | 85 | 9 | 44 | 16 | 83 | 15 | 36 | 97 | 99 | 8 | 21 | 73 | 71 |
| 9 | 42 | 14 | 56 | 12 | 42 | 14 | 56 | 18 | 7 | 44 | 16 | 10 | 49 | 0 | 68 |
| 10 | 49 | 0 | 68 | 13 | 76 | 2 | 78 | 30 | 4 | 83 | 16 | 12 | 6 | 87 | 30 |
| 11 | 14 | 42 | 41 | 14 | 19 | 29 | 69 | 31 | 2 | 76 | 75 | 13 | 76 | 2 | 78 |
| 12 | 4 | 83 | 16 | 15 | 1 | 56 | 14 | 32 | 22 | 55 | 74 | 14 | 2 | 76 | 75 |
| 14 | 28 | 23 | 53 | 16 | 49 | 75 | 76 | 33 | 31 | 89 | 62 | 16 | 21 | 40 | 74 |
| 15 | 11 | 79 | 67 | 18 | 35 | 48 | 63 | 36 | 4 | 59 | 82 | 17 | 59 | 4 | 83 |
| 16 | 22 | 55 | 74 | 19 | 40 | 21 | 73 | 37 | 44 | 16 | 83 | 18 | 4 | 59 | 82 |
| 18 | 24 | 54 | 53 | 20 | 14 | 42 | 41 | 38 | 14 | 42 | 41 | 20 | 14 | 42 | 41 |
| 19 | 4 | 59 | 82 | 21 | 0 | 49 | 75 | 39 | 59 | 4 | 83 | 22 | 4 | 83 | 16 |
| 22 | 44 | 16 | 83 | 23 | 69 | 30 | 87 | 41 | 6 | 87 | 30 | 24 | 60 | 15 | 85 |
| 23 | 20 | 72 | 71 | 25 | 6 | 81 | 47 | 42 | 49 | 0 | 68 | 26 | 0 | 68 | 26 |
| 25 | 40 | 21 | 73 | 26 | 55 | 22 | 66 | 45 | 93 | 94 | 96 | 27 | 24 | 54 | 53 |
| | | | | | | | | | | | | | | | |
| 172 | 57 | 12 | 93 | 238 | 13 | 41 | 42 | 425 | 15 | 60 | 84 | 212 | 18 | 10 | 61 |
| 175 | 56 | 1 | 86 | 240 | 9 | 61 | 10 | 468 | 1 | 86 | 96 | 213 | 56 | 1 | 86 |
| 185 | 31 | 9 | 61 | 250 | 88 | 5 | 95 | 470 | 92 | 98 | 95 | 215 | 15 | 85 | 37 |
| 201 | 35 | 46 | 45 | 267 | 3 | 38 | 66 | 513 | 91 | 58 | 92 | 224 | 17 | 51 | 88 |
| 202 | 40 | 74 | 55 | 306 | 12 | 93 | 94 | 530 | 13 | 43 | 37 | 246 | 57 | 12 | 93 |
| 203 | 51 | 17 | 82 | 317 | 18 | 47 | 81 | 539 | 39 | 20 | 72 | 247 | 1 | 86 | 96 |
| 212 | 90 | 99 | 97 | 321 | 35 | 46 | 45 | 571 | 25 | 11 | 79 | 248 | 18 | 47 | 81 |
| 213 | 18 | 47 | 81 | 322 | 24 | 3 | 38 | 720 | 57 | 12 | 93 | 280 | 43 | 37 | 85 |
| 225 | 12 | 93 | 94 | 331 | 60 | 84 | 90 | 721 | 12 | 93 | 94 | 291 | 13 | 41 | 42 |
| 226 | 13 | 41 | 42 | 336 | 48 | 63 | 62 | 750 | 81 | 6 | 87 | 295 | 25 | 27 | 52 |
| 227 | 13 | 43 | 37 | 387 | 91 | 58 | 92 | 802 | 25 | 27 | 52 | 328 | 91 | 58 | 92 |
| 228 | 25 | 27 | 52 | 394 | 1 | 86 | 96 | 953 | 7 | 45 | 46 | 329 | 7 | 45 | 46 |
| 229 | 43 | 37 | 85 | 396 | 7 | 45 | 46 | 1289 | 17 | 51 | 88 | 351 | 35 | 46 | 45 |
| 263 | 7 | 45 | 46 | 425 | 8 | 80 | 77 | 1476 | 15 | 85 | 37 | 387 | 13 | 43 | 37 |
| 264 | 8 | 50 | 32 | 427 | 64 | 70 | 65 | 1674 | 3 | 38 | 66 | 403 | 8 | 80 | 77 |
| 265 | 24 | 3 | 38 | 464 | 50 | 32 | 78 | 1815 | 23 | 53 | 54 | 468 | 41 | 13 | 43 |
| 280 | 10 | 18 | 47 | 465 | 19 | 65 | 70 | 2419 | 8 | 80 | 77 | 505 | 50 | 32 | 78 |
| 298 | 58 | 92 | 98 | 521 | 18 | 10 | 61 | 2596 | 50 | 32 | 78 | 517 | 24 | 3 | 38 |
| 391 | 41 | 13 | 43 | 539 | 10 | 18 | 47 | 2600 | 58 | 92 | 98 | 596 | 10 | 18 | 47 |
| 540 | 8 | 80 | 77 | 698 | 58 | 92 | 98 | 2602 | 86 | 96 | 94 | 608 | 58 | 92 | 98 |

**Table 4.13:** Table of Top 20, Bottom 20 Global 5-grams for the EIL101

| EIL101 Top 20, Bottom 20 Global 5-grams | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RILO** | | | | | | | **NNLO** | | | | | | **ACO-LO** | | | | | |
| rank | edge | | | | | | rank | edge | | | | | rank | edge | | | | |
| 2 | 38 | 66 | 22 | 55 | 74 | | 2 | 38 | 66 | 22 | 55 | 74 | 1 | 71 | 73 | 21 | 40 | 74 |
| 4 | 1 | 56 | 14 | 42 | 41 | | 5 | 19 | 29 | 69 | 30 | 87 | 3 | 1 | 56 | 14 | 42 | 41 |
| 5 | 24 | 54 | 53 | 23 | 28 | | 6 | 1 | 56 | 14 | 42 | 41 | 8 | 38 | 66 | 22 | 55 | 74 |
| 7 | 16 | 83 | 4 | 59 | 82 | | 13 | 49 | 75 | 76 | 2 | 78 | 14 | 16 | 83 | 4 | 59 | 82 |
| 9 | 11 | 79 | 67 | 28 | 23 | | 14 | 37 | 85 | 15 | 60 | 84 | 15 | 68 | 0 | 49 | 75 | 76 |
| 12 | 53 | 23 | 28 | 67 | 79 | | 18 | 43 | 37 | 85 | 15 | 60 | 20 | 0 | 49 | 75 | 76 | 2 |
| 14 | 0 | 49 | 75 | 76 | 2 | | 19 | 0 | 49 | 75 | 76 | 2 | 21 | 49 | 75 | 76 | 2 | 78 |
| 16 | 49 | 75 | 76 | 2 | 78 | | 22 | 45 | 7 | 44 | 16 | 83 | 24 | 55 | 74 | 40 | 21 | 73 |
| 17 | 68 | 0 | 49 | 75 | 76 | | 24 | 13 | 43 | 37 | 85 | 15 | 25 | 21 | 40 | 74 | 55 | 22 |
| 18 | 70 | 64 | 34 | 33 | 77 | | 27 | 26 | 68 | 0 | 49 | 75 | 26 | 40 | 74 | 55 | 22 | 66 |
| 23 | 19 | 65 | 70 | 64 | 34 | | 29 | 4 | 83 | 16 | 44 | 7 | 27 | 26 | 68 | 0 | 49 | 75 |
| 29 | 31 | 89 | 62 | 63 | 48 | | 35 | 16 | 83 | 4 | 59 | 82 | 33 | 70 | 64 | 34 | 33 | 77 |
| 30 | 44 | 16 | 83 | 4 | 59 | | 40 | 17 | 82 | 59 | 4 | 83 | 35 | 40 | 21 | 73 | 71 | 72 |
| 31 | 25 | 11 | 79 | 67 | 28 | | 42 | 32 | 78 | 2 | 76 | 75 | 39 | 24 | 54 | 53 | 23 | 28 |
| 32 | 17 | 82 | 59 | 4 | 83 | | 44 | 44 | 16 | 83 | 4 | 59 | 41 | 20 | 72 | 71 | 73 | 21 |
| 38 | 35 | 48 | 63 | 62 | 89 | | 56 | 55 | 74 | 40 | 21 | 73 | 42 | 11 | 79 | 67 | 28 | 23 |
| 39 | 32 | 78 | 2 | 76 | 75 | | 59 | 40 | 74 | 55 | 22 | 66 | 43 | 4 | 83 | 16 | 44 | 7 |
| 40 | 54 | 53 | 23 | 28 | 67 | | 62 | 71 | 73 | 21 | 40 | 74 | 45 | 17 | 82 | 59 | 4 | 83 |
| 41 | 20 | 72 | 71 | 73 | 21 | | 67 | 39 | 20 | 72 | 71 | 73 | 55 | 53 | 23 | 28 | 67 | 79 |
| 45 | 40 | 21 | 73 | 71 | 72 | | 68 | 21 | 40 | 74 | 55 | 22 | 58 | 6 | 87 | 30 | 69 | 29 |
| | | | | | | | | | | | | | | | | | | |
| 464 | 12 | 93 | 94 | 96 | 86 | | 833 | 64 | 34 | 33 | 77 | 80 | 605 | 78 | 32 | 50 | 8 | 80 |
| 465 | 1 | 86 | 96 | 94 | 93 | | 834 | 33 | 34 | 64 | 70 | 65 | 613 | 60 | 84 | 90 | 99 | 97 |
| 613 | 20 | 39 | 57 | 12 | 93 | | 835 | 19 | 65 | 70 | 64 | 34 | 628 | 38 | 3 | 24 | 54 | 53 |
| 623 | 51 | 88 | 5 | 95 | 98 | | 840 | 18 | 47 | 81 | 6 | 87 | 629 | 54 | 24 | 3 | 38 | 66 |
| 633 | 37 | 43 | 13 | 41 | 42 | | 872 | 29 | 19 | 65 | 70 | 64 | 630 | 22 | 66 | 38 | 3 | 24 |
| 634 | 14 | 42 | 41 | 13 | 43 | | 882 | 14 | 56 | 1 | 86 | 96 | 687 | 14 | 56 | 1 | 86 | 96 |
| 649 | 52 | 27 | 25 | 11 | 79 | | 883 | 1 | 86 | 96 | 94 | 93 | 693 | 14 | 42 | 41 | 13 | 43 |
| 783 | 39 | 57 | 12 | 93 | 94 | | 889 | 49 | 0 | 68 | 26 | 100 | 702 | 92 | 58 | 91 | 36 | 97 |
| 845 | 32 | 50 | 8 | 80 | 77 | | 1124 | 51 | 88 | 5 | 95 | 98 | 776 | 12 | 93 | 94 | 96 | 86 |
| 846 | 33 | 77 | 80 | 8 | 50 | | 1160 | 50 | 32 | 78 | 2 | 76 | 813 | 20 | 39 | 57 | 12 | 93 |
| 847 | 8 | 80 | 77 | 33 | 34 | | 1186 | 69 | 29 | 19 | 65 | 70 | 829 | 41 | 13 | 43 | 37 | 85 |
| 848 | 30 | 69 | 29 | 19 | 65 | | 1215 | 5 | 95 | 98 | 92 | 58 | 841 | 91 | 58 | 92 | 98 | 95 |
| 849 | 6 | 81 | 47 | 18 | 10 | | 1216 | 91 | 58 | 92 | 98 | 95 | 890 | 18 | 47 | 81 | 6 | 87 |
| 850 | 61 | 10 | 18 | 47 | 81 | | 1247 | 6 | 81 | 47 | 18 | 10 | 894 | 50 | 32 | 78 | 2 | 76 |
| 851 | 9 | 61 | 10 | 18 | 47 | | 1289 | 39 | 57 | 12 | 93 | 94 | 902 | 88 | 5 | 95 | 98 | 92 |
| 893 | 88 | 5 | 95 | 98 | 92 | | 1396 | 88 | 5 | 95 | 98 | 92 | 1210 | 37 | 43 | 13 | 41 | 42 |
| 896 | 16 | 44 | 7 | 45 | 46 | | 2525 | 60 | 84 | 90 | 99 | 97 | 1233 | 5 | 95 | 98 | 92 | 58 |
| 903 | 17 | 51 | 88 | 5 | 95 | | 3169 | 10 | 61 | 9 | 31 | 89 | 1325 | 36 | 91 | 58 | 92 | 98 |
| 1451 | 14 | 56 | 1 | 86 | 96 | | 3170 | 18 | 10 | 61 | 9 | 31 | 1340 | 57 | 12 | 93 | 94 | 96 |
| 3960 | 57 | 12 | 93 | 94 | 96 | | 3171 | 9 | 61 | 10 | 18 | 47 | 3057 | 6 | 81 | 47 | 18 | 10 |

**Table 4.14:** Table of the Top 40 Bigrams for the EIL101.
Edge (24,54) is shown in the edge column as | 24 | 54 |. Highlighted cells are global edges.

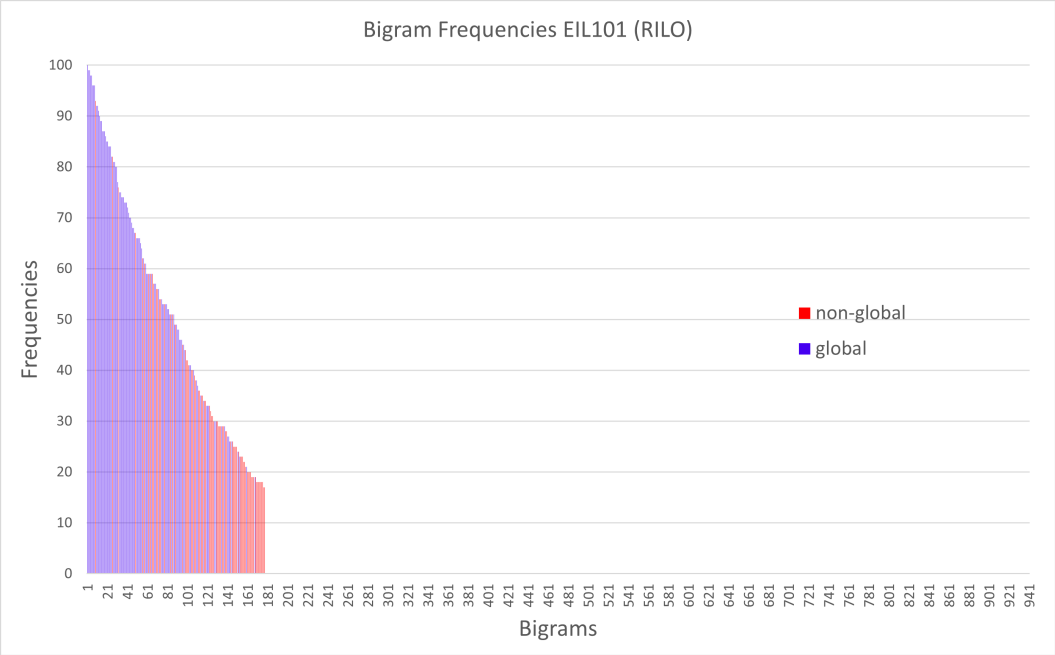
| EIL101 Top 40 Bigrams | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RILO | | | NNLO | | | ACO-S1 | | | ACO-LO | | | |
| edge | | freq | edge | | freq | edge | | freq | edge | | freq | |
| 24 | 54 | 100 | 24 | 54 | 100 | 24 | 54 | 99 | 24 | 54 | 100 | |
| 31 | 89 | 99 | 33 | 77 | 99 | 36 | 97 | 95 | 14 | 42 | 99 | |
| 33 | 77 | 99 | 22 | 66 | 99 | 33 | 77 | 95 | 33 | 77 | 98 | |
| 14 | 42 | 98 | 23 | 28 | 98 | 21 | 73 | 95 | 23 | 28 | 98 | |
| 23 | 28 | 98 | 14 | 42 | 97 | 23 | 28 | 93 | 62 | 89 | 97 | |
| 21 | 40 | 96 | 31 | 89 | 97 | 14 | 56 | 92 | 35 | 48 | 96 | |
| 35 | 48 | 96 | 0 | 68 | 96 | 37 | 43 | 92 | 31 | 89 | 96 | |
| 67 | 79 | 96 | 21 | 40 | 96 | 67 | 79 | 92 | 21 | 40 | 96 | |
| 32 | 80 | 93 | 67 | 79 | 95 | 48 | 63 | 92 | 38 | 66 | 96 | |
| 22 | 66 | 92 | 35 | 48 | 94 | 21 | 40 | 92 | 0 | 68 | 93 | |
| 62 | 89 | 92 | 1 | 56 | 93 | 13 | 37 | 91 | 67 | 79 | 93 | |
| 36 | 97 | 91 | 15 | 60 | 92 | 91 | 96 | 90 | 48 | 63 | 91 | |
| 20 | 72 | 90 | 95 | 98 | 90 | 14 | 42 | 90 | 32 | 80 | 91 | |
| 0 | 68 | 89 | 32 | 80 | 90 | 24 | 38 | 89 | 95 | 98 | 91 | |
| 39 | 57 | 89 | 36 | 97 | 89 | 1 | 56 | 88 | 39 | 57 | 91 | |
| 15 | 85 | 87 | 5 | 88 | 89 | 46 | 47 | 88 | 29 | 69 | 89 | |
| 30 | 87 | 87 | 38 | 66 | 89 | 55 | 74 | 88 | 20 | 72 | 89 | |
| 38 | 66 | 87 | 16 | 83 | 88 | 5 | 93 | 88 | 14 | 56 | 89 | |
| 52 | 100 | 86 | 30 | 87 | 87 | 39 | 57 | 87 | 64 | 70 | 88 | |
| 16 | 83 | 85 | 90 | 99 | 87 | 59 | 82 | 86 | 13 | 37 | 87 | |
| 95 | 98 | 85 | 37 | 85 | 86 | 0 | 68 | 86 | 30 | 87 | 87 | |
| 2 | 76 | 84 | 29 | 69 | 85 | 38 | 66 | 86 | 52 | 100 | 87 | |
| 2 | 78 | 84 | 15 | 85 | 85 | 32 | 80 | 84 | 2 | 76 | 86 | |
| 29 | 69 | 84 | 52 | 100 | 84 | 31 | 89 | 84 | 21 | 73 | 86 | |
| 1 | 56 | 82 | 5 | 93 | 83 | 64 | 70 | 84 | 4 | 83 | 85 | |
| 13 | 37 | 82 | 2 | 78 | 83 | 33 | 34 | 83 | 33 | 34 | 85 | |
| 58 | 91 | 81 | 64 | 65 | 81 | 30 | 87 | 83 | 36 | 97 | 85 | |
| 75 | 76 | 81 | 64 | 70 | 81 | 7 | 44 | 83 | 59 | 82 | 84 | |
| 64 | 70 | 80 | 34 | 70 | 80 | 95 | 98 | 83 | 47 | 81 | 83 | |
| 86 | 96 | 80 | 7 | 45 | 79 | 58 | 98 | 82 | 71 | 73 | 83 | |
| 59 | 82 | 77 | 6 | 81 | 78 | 84 | 90 | 82 | 22 | 66 | 83 | |
| 5 | 93 | 76 | 2 | 76 | 77 | 2 | 76 | 82 | 9 | 61 | 81 | |
| 0 | 49 | 75 | 8 | 50 | 77 | 4 | 83 | 82 | 1 | 56 | 80 | |
| 4 | 83 | 75 | 13 | 43 | 77 | 47 | 81 | 82 | 8 | 50 | 79 | |
| 10 | 18 | 74 | 49 | 75 | 76 | 20 | 72 | 81 | 5 | 88 | 79 | |
| 15 | 60 | 74 | 10 | 63 | 76 | 35 | 48 | 80 | 58 | 91 | 79 | |
| 19 | 65 | 74 | 20 | 72 | 75 | 20 | 71 | 79 | 37 | 43 | 78 | |
| 22 | 55 | 73 | 19 | 29 | 75 | 7 | 81 | 78 | 15 | 85 | 78 | |
| 33 | 34 | 73 | 4 | 83 | 75 | 71 | 73 | 78 | 5 | 93 | 78 | |
| 48 | 63 | 73 | 16 | 44 | 75 | 5 | 88 | 77 | 19 | 65 | 76 | |

**Table 4.15:** Table of Top 40 Trigrams for the EIL101. Highlighted cells are global edges.

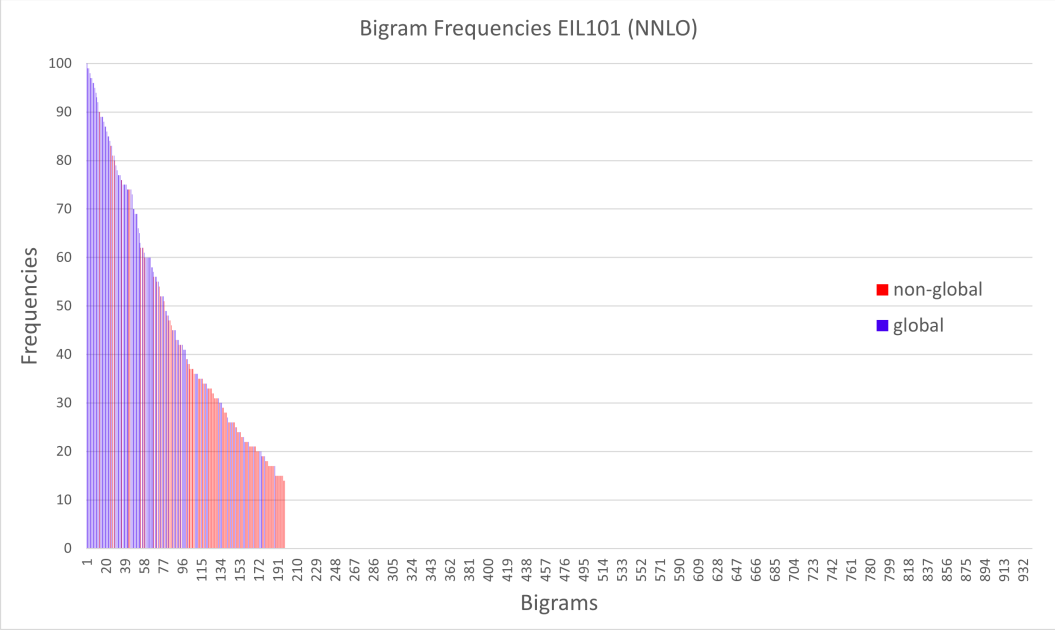
| EIL101 Top 40 Trigrams | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RILO | | | | NNLO | | | | ACO-S1 | | | | ACO-LO | | | | |
| edge | | | freq | edge | | | freq | edge | | | freq | edge | | | freq | |
| 31 | 89 | 62 | 91 | 22 | 66 | 38 | 88 | 40 | 21 | 73 | 88 | 31 | 89 | 62 | 93 | |
| 22 | 66 | 38 | 80 | 60 | 15 | 85 | 78 | 38 | 24 | 54 | 88 | 42 | 14 | 56 | 88 | |
| 34 | 33 | 77 | 72 | 15 | 85 | 37 | 72 | 1 | 56 | 14 | 83 | 35 | 48 | 63 | 87 | |
| 35 | 48 | 63 | 69 | 88 | 5 | 93 | 72 | 13 | 37 | 43 | 82 | 34 | 33 | 77 | 83 | |
| 76 | 2 | 78 | 68 | 31 | 89 | 62 | 71 | 42 | 14 | 56 | 82 | 40 | 21 | 73 | 82 | |
| 2 | 76 | 75 | 65 | 49 | 0 | 68 | 69 | 34 | 33 | 77 | 78 | 22 | 66 | 38 | 78 | |
| 55 | 22 | 66 | 65 | 34 | 33 | 77 | 64 | 24 | 38 | 66 | 77 | 1 | 56 | 14 | 70 | |
| 60 | 15 | 85 | 65 | 4 | 83 | 16 | 64 | 46 | 47 | 81 | 74 | 21 | 73 | 71 | 69 | |
| 42 | 14 | 56 | 64 | 44 | 16 | 83 | 64 | 21 | 73 | 71 | 74 | 13 | 37 | 43 | 67 | |
| 49 | 0 | 68 | 64 | 65 | 64 | 70 | 63 | 35 | 48 | 63 | 72 | 49 | 0 | 68 | 66 | |
| 14 | 42 | 41 | 63 | 34 | 70 | 64 | 63 | 15 | 43 | 37 | 70 | 45 | 35 | 48 | 65 | |
| 4 | 83 | 16 | 60 | 42 | 14 | 56 | 63 | 0 | 68 | 26 | 68 | 6 | 87 | 30 | 63 | |
| 38 | 24 | 54 | 59 | 76 | 2 | 78 | 60 | 71 | 20 | 72 | 68 | 76 | 2 | 78 | 62 | |
| 28 | 23 | 53 | 58 | 19 | 29 | 69 | 59 | 58 | 91 | 96 | 67 | 2 | 76 | 75 | 61 | |
| 11 | 79 | 67 | 57 | 1 | 56 | 14 | 59 | 36 | 97 | 99 | 66 | 46 | 47 | 81 | 60 | |
| 22 | 55 | 74 | 57 | 49 | 75 | 76 | 57 | 44 | 7 | 81 | 66 | 21 | 40 | 74 | 60 | |
| 24 | 38 | 66 | 54 | 38 | 24 | 54 | 56 | 58 | 98 | 95 | 66 | 59 | 4 | 83 | 59 | |
| 24 | 54 | 53 | 54 | 35 | 48 | 63 | 54 | 7 | 44 | 16 | 65 | 4 | 59 | 82 | 59 | |
| 4 | 59 | 82 | 53 | 40 | 21 | 73 | 52 | 88 | 5 | 93 | 65 | 38 | 24 | 54 | 59 | |
| 97 | 36 | 99 | 53 | 14 | 42 | 41 | 52 | 5 | 93 | 94 | 65 | 14 | 42 | 41 | 59 | |
| 39 | 57 | 52 | 52 | 0 | 49 | 75 | 51 | 56 | 1 | 72 | 64 | 88 | 5 | 93 | 58 | |
| 44 | 16 | 83 | 52 | 24 | 38 | 66 | 51 | 20 | 71 | 73 | 64 | 4 | 83 | 16 | 57 | |
| 20 | 72 | 71 | 51 | 69 | 30 | 87 | 51 | 23 | 28 | 78 | 63 | 24 | 38 | 66 | 56 | |
| 57 | 52 | 100 | 50 | 33 | 77 | 78 | 50 | 1 | 72 | 20 | 63 | 60 | 15 | 85 | 55 | |
| 40 | 21 | 73 | 50 | 6 | 81 | 47 | 50 | 91 | 58 | 98 | 62 | 33 | 77 | 78 | 55 | |
| 59 | 4 | 83 | 50 | 55 | 22 | 66 | 50 | 7 | 81 | 47 | 62 | 0 | 68 | 26 | 55 | |
| 1 | 56 | 14 | 49 | 2 | 76 | 75 | 49 | 91 | 96 | 94 | 61 | 24 | 54 | 53 | 55 | |
| 5 | 93 | 12 | 49 | 46 | 47 | 81 | 48 | 92 | 36 | 97 | 57 | 10 | 63 | 48 | 54 | |
| 25 | 39 | 57 | 49 | 5 | 93 | 94 | 48 | 33 | 77 | 78 | 56 | 8 | 80 | 32 | 53 | |
| 0 | 68 | 26 | 48 | 22 | 55 | 74 | 48 | 4 | 83 | 16 | 56 | 65 | 64 | 70 | 53 | |
| 33 | 77 | 78 | 48 | 33 | 34 | 70 | 47 | 2 | 76 | 75 | 55 | 34 | 70 | 64 | 53 | |
| 45 | 35 | 48 | 48 | 29 | 69 | 30 | 47 | 22 | 55 | 74 | 53 | 33 | 34 | 70 | 50 | |
| 15 | 85 | 37 | 48 | 45 | 35 | 48 | 47 | 31 | 89 | 62 | 51 | 58 | 91 | 96 | 49 | |
| 6 | 87 | 30 | 47 | 26 | 100 | 52 | 47 | 33 | 34 | 70 | 50 | 49 | 75 | 76 | 49 | |
| 46 | 35 | 48 | 47 | 57 | 52 | 100 | 46 | 12 | 86 | 41 | 50 | 19 | 29 | 69 | 49 | |
| 72 | 71 | 73 | 47 | 46 | 35 | 48 | 46 | 4 | 59 | 82 | 50 | 55 | 22 | 66 | 48 | |
| 8 | 80 | 32 | 44 | 0 | 68 | 26 | 42 | 44 | 16 | 83 | 49 | 10 | 62 | 89 | 48 | |
| 86 | 96 | 94 | 44 | 86 | 96 | 91 | 42 | 14 | 42 | 41 | 49 | 19 | 65 | 64 | 47 | |
| 36 | 99 | 90 | 43 | 19 | 65 | 64 | 42 | 59 | 4 | 83 | 49 | 7 | 81 | 47 | 46 | |
| 46 | 47 | 81 | 43 | 53 | 79 | 67 | 41 | 5 | 88 | 17 | 49 | 37 | 43 | 85 | 45 | |

**Table 4.16:** Table of Top 40 EIL101 5-grams. Highlighted cells are global edges.

| EIL101 Top 40 5-grams | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RILO | | | | | | | NNLO | | | | | | ACO-LO | | | | | |
| edge | | | | | freq | | edge | | | | | freq | edge | | | | | freq |
| 22 | 66 | 38 | 24 | 54 | 52 | | 22 | 66 | 38 | 24 | 54 | 51 | 71 | 73 | 21 | 40 | 74 | 56 |
| 38 | 66 | 22 | 55 | 74 | 51 | | 38 | 66 | 22 | 55 | 74 | 46 | 22 | 66 | 38 | 24 | 54 | 53 |
| 24 | 38 | 66 | 22 | 55 | 43 | | 19 | 65 | 64 | 70 | 34 | 40 | 1 | 56 | 14 | 42 | 41 | 47 |
| 1 | 56 | 14 | 42 | 41 | 37 | | 68 | 0 | 49 | 75 | 76 | 39 | 33 | 34 | 70 | 64 | 65 | 45 |
| 24 | 54 | 53 | 23 | 28 | 35 | | 19 | 29 | 69 | 30 | 87 | 38 | 64 | 70 | 34 | 33 | 77 | 44 |
| 19 | 65 | 64 | 70 | 34 | 34 | | 1 | 56 | 14 | 42 | 41 | 36 | 19 | 65 | 64 | 70 | 34 | 42 |
| 16 | 83 | 4 | 59 | 82 | 34 | | 24 | 38 | 66 | 22 | 55 | 35 | 13 | 37 | 43 | 85 | 15 | 40 |
| 33 | 77 | 78 | 2 | 76 | 30 | | 33 | 34 | 70 | 64 | 65 | 33 | 38 | 66 | 22 | 55 | 74 | 40 |
| 11 | 79 | 67 | 28 | 23 | 30 | | 3 | 54 | 24 | 38 | 66 | 33 | 56 | 14 | 42 | 41 | 86 | 40 |
| 56 | 14 | 42 | 41 | 86 | 29 | | 40 | 21 | 74 | 22 | 66 | 33 | 10 | 63 | 48 | 35 | 45 | 36 |
| 25 | 39 | 57 | 52 | 100 | 28 | | 64 | 70 | 34 | 33 | 77 | 32 | 24 | 38 | 66 | 22 | 55 | 36 |
| 53 | 23 | 28 | 67 | 79 | 28 | | 21 | 74 | 22 | 66 | 38 | 32 | 37 | 43 | 85 | 15 | 60 | 34 |
| 15 | 85 | 37 | 13 | 43 | 27 | | 49 | 75 | 76 | 2 | 78 | 32 | 1 | 72 | 20 | 71 | 73 | 33 |
| 0 | 49 | 75 | 76 | 2 | 27 | | 37 | 85 | 15 | 60 | 84 | 32 | 16 | 83 | 4 | 59 | 82 | 32 |
| 14 | 42 | 41 | 86 | 96 | 26 | | 85 | 37 | 13 | 43 | 90 | 31 | 68 | 0 | 49 | 75 | 76 | 32 |
| 49 | 75 | 76 | 2 | 78 | 26 | | 15 | 85 | 37 | 13 | 43 | 31 | 42 | 14 | 56 | 1 | 72 | 32 |
| 68 | 0 | 49 | 75 | 76 | 26 | | 13 | 37 | 85 | 15 | 60 | 31 | 21 | 73 | 71 | 20 | 72 | 31 |
| 70 | 64 | 34 | 33 | 77 | 25 | | 43 | 37 | 85 | 15 | 60 | 31 | 22 | 74 | 40 | 21 | 73 | 30 |
| 75 | 76 | 2 | 78 | 77 | 24 | | 0 | 49 | 75 | 76 | 2 | 30 | 20 | 71 | 73 | 21 | 40 | 30 |
| 2 | 78 | 77 | 33 | 34 | 24 | | 76 | 2 | 78 | 32 | 80 | 30 | 0 | 49 | 75 | 76 | 2 | 29 |
| 85 | 37 | 13 | 43 | 90 | 24 | | 8 | 34 | 70 | 64 | 65 | 30 | 49 | 75 | 76 | 2 | 78 | 29 |
| 13 | 37 | 85 | 15 | 60 | 23 | | 45 | 7 | 44 | 16 | 83 | 29 | 33 | 77 | 78 | 2 | 76 | 29 |
| 19 | 65 | 70 | 64 | 34 | 22 | | 37 | 13 | 43 | 90 | 99 | 28 | 31 | 89 | 62 | 10 | 63 | 29 |
| 3 | 54 | 24 | 38 | 66 | 22 | | 13 | 43 | 37 | 85 | 15 | 28 | 55 | 74 | 40 | 21 | 73 | 28 |
| 62 | 89 | 31 | 29 | 69 | 22 | | 85 | 15 | 60 | 84 | 92 | 26 | 21 | 40 | 74 | 55 | 22 | 28 |
| 64 | 70 | 34 | 33 | 77 | 22 | | 56 | 14 | 42 | 41 | 86 | 25 | 40 | 74 | 55 | 22 | 66 | 28 |
| 33 | 34 | 70 | 64 | 65 | 22 | | 26 | 68 | 0 | 49 | 75 | 24 | 26 | 68 | 0 | 49 | 75 | 28 |
| 76 | 2 | 78 | 32 | 80 | 22 | | 22 | 74 | 21 | 40 | 72 | 24 | 14 | 42 | 41 | 86 | 96 | 28 |
| 31 | 89 | 62 | 63 | 48 | 21 | | 4 | 83 | 16 | 44 | 7 | 24 | 70 | 34 | 33 | 77 | 78 | 27 |
| 44 | 16 | 83 | 4 | 59 | 21 | | 42 | 13 | 43 | 37 | 85 | 24 | 2 | 78 | 77 | 33 | 34 | 27 |
| 25 | 11 | 79 | 67 | 28 | 20 | | 14 | 42 | 41 | 86 | 96 | 23 | 48 | 63 | 10 | 62 | 89 | 27 |
| 17 | 82 | 59 | 4 | 83 | 20 | | 55 | 38 | 66 | 22 | 74 | 23 | 56 | 1 | 72 | 20 | 71 | 27 |
| 13 | 37 | 43 | 85 | 15 | 20 | | 23 | 28 | 77 | 33 | 34 | 23 | 70 | 64 | 34 | 33 | 77 | 27 |
| 18 | 10 | 63 | 48 | 35 | 19 | | 8 | 50 | 80 | 32 | 78 | 23 | 19 | 50 | 8 | 80 | 32 | 26 |
| 27 | 100 | 52 | 57 | 39 | 19 | | 16 | 83 | 4 | 59 | 82 | 23 | 40 | 21 | 73 | 71 | 72 | 26 |
| 53 | 54 | 24 | 38 | 66 | 19 | | 27 | 26 | 68 | 0 | 49 | 22 | 35 | 48 | 63 | 10 | 62 | 26 |
| 64 | 34 | 33 | 77 | 78 | 19 | | 70 | 34 | 33 | 77 | 78 | 22 | 14 | 56 | 1 | 72 | 20 | 26 |
| 35 | 48 | 63 | 62 | 89 | 19 | | 44 | 7 | 45 | 35 | 48 | 22 | 7 | 45 | 35 | 48 | 63 | 25 |
| 32 | 78 | 2 | 76 | 75 | 19 | | 14 | 42 | 13 | 43 | 37 | 22 | 24 | 54 | 53 | 23 | 28 | 25 |
| 54 | 53 | 23 | 28 | 67 | 19 | | 17 | 82 | 59 | 4 | 83 | 22 | 14 | 42 | 13 | 37 | 43 | 24 |

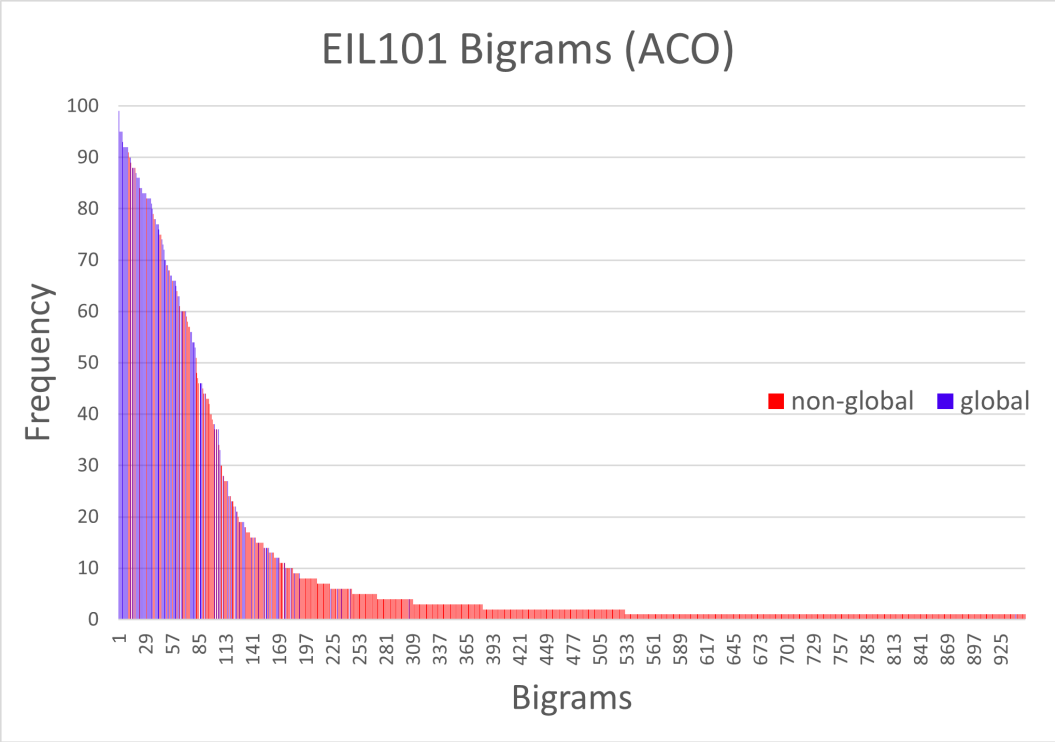



**(a)** EIL101 Bigrams RILO

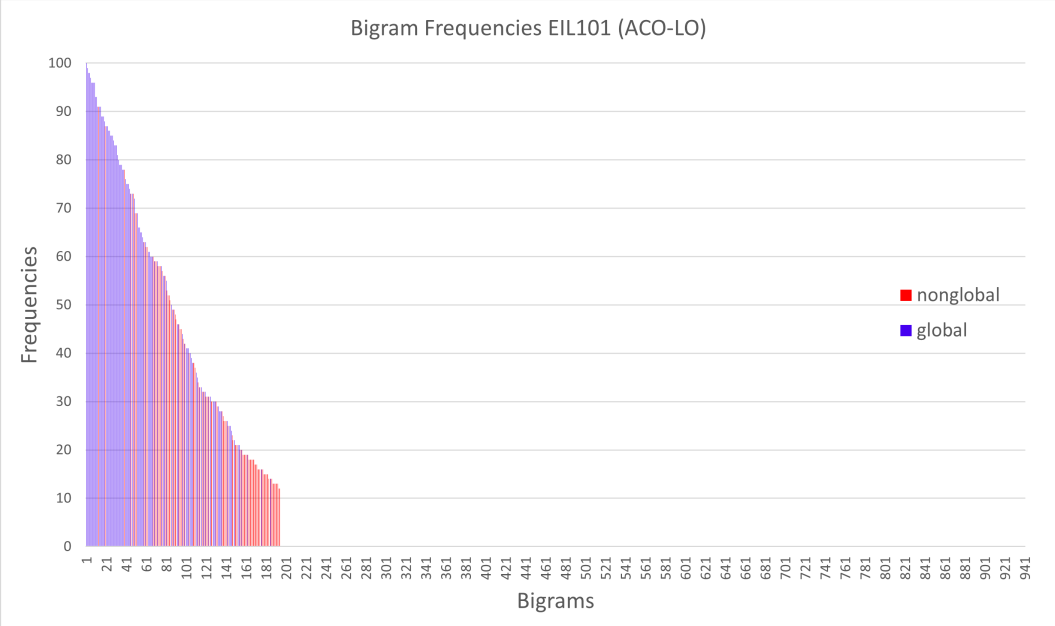



**(b)** EIL101 Bigrams NNLO

**Figure 4.7:** Bigram Frequencies for the EIL101 vs. the Number of Bigrams (a) RILO (b) NNLO

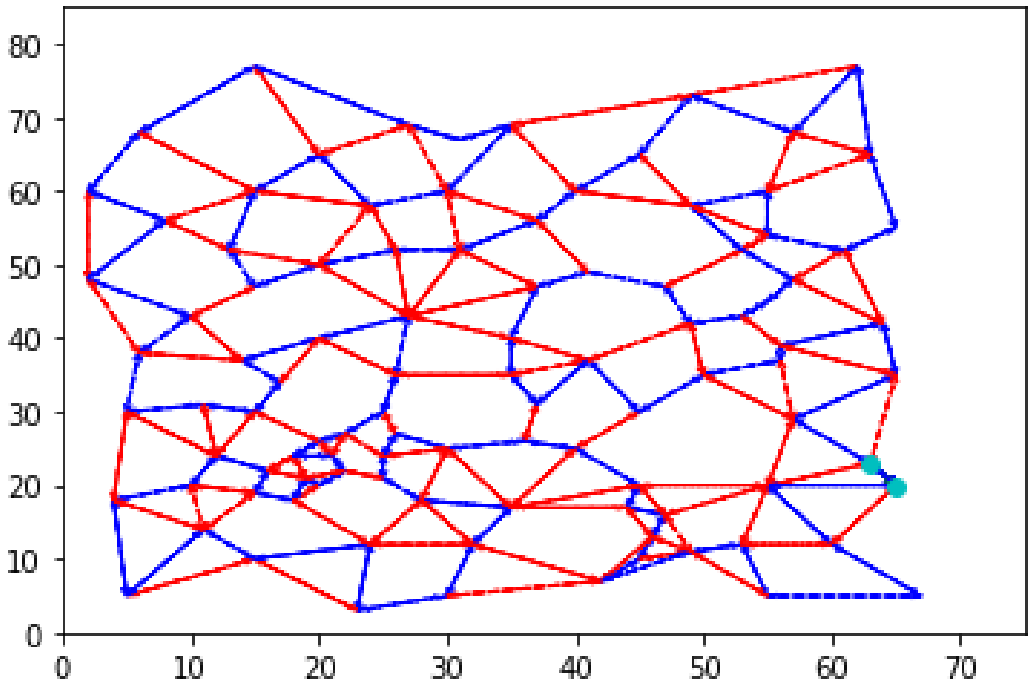



**(a)** EIL101 Bigrams ACO-S1

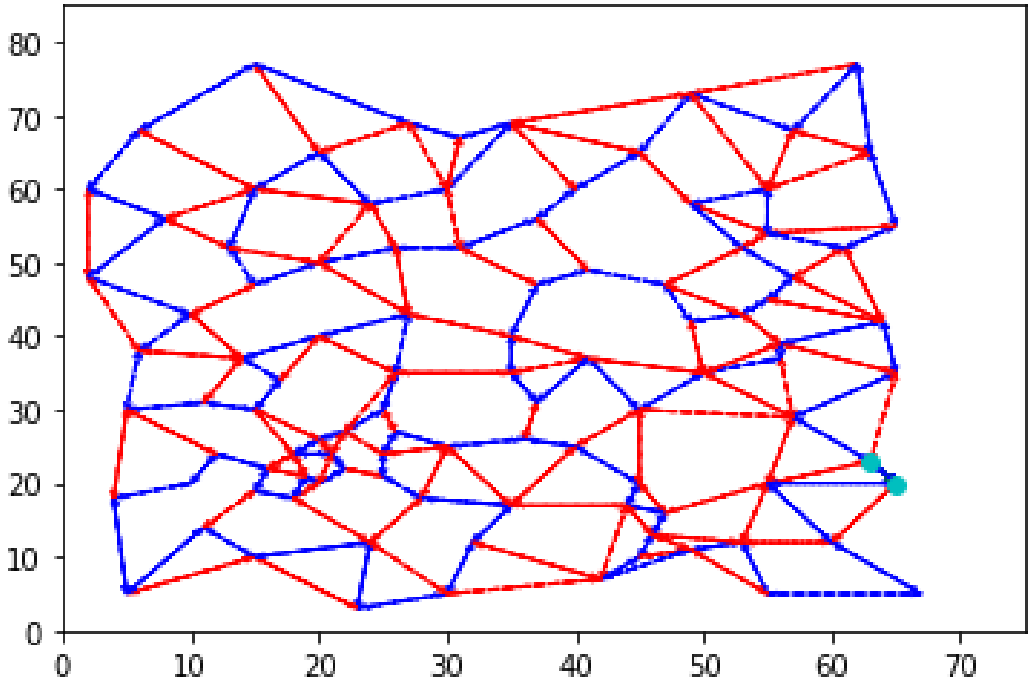



**(b)** EIL101 Bigrams ACO-LO

**Figure 4.8:** Bigram Frequencies for the EIL101 vs. the Number of Bigrams (a) ACO-S1 (b) ACO-LO

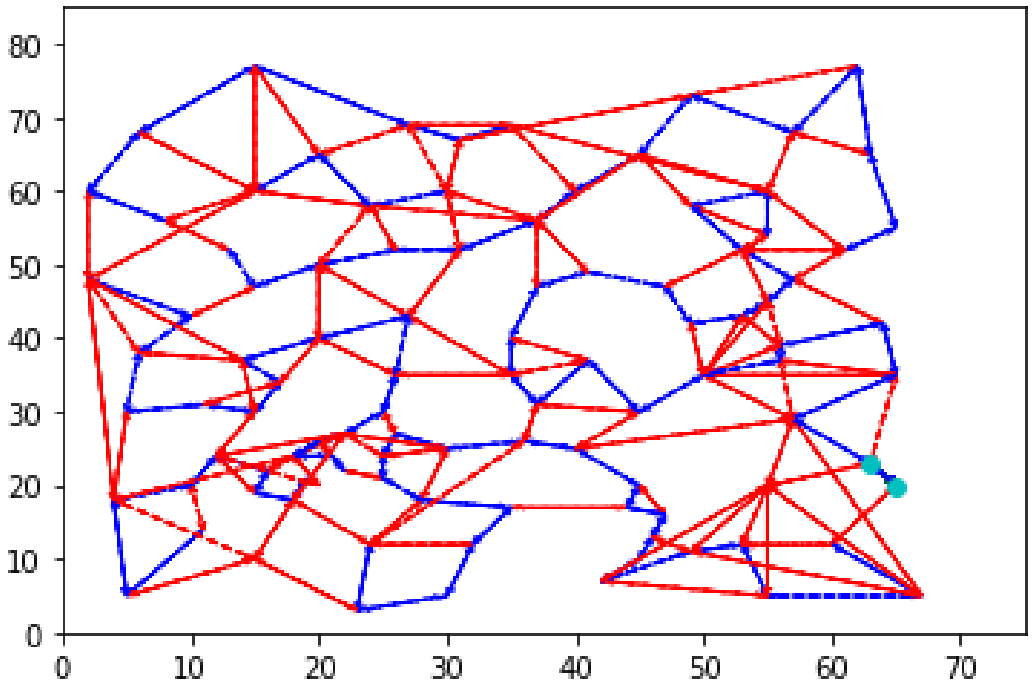



**(a)** EIL101 Top 200 Bigrams RILO

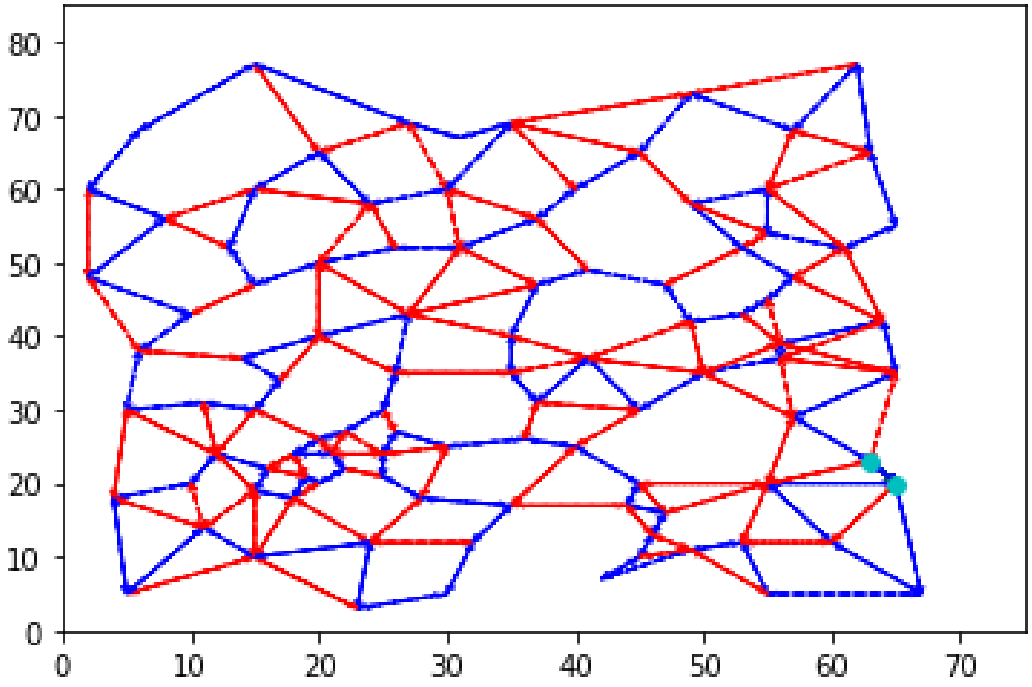



**(b)** EIL101 Top 200 Bigrams NNLO

**Figure 4.9:** Top 200 EIL101 Bigrams (a) RILO (b) NNLO
Note: blue edges are globally optimal edges while red are locally optimal edges. The edge with teal vertices is the first ranked edge.

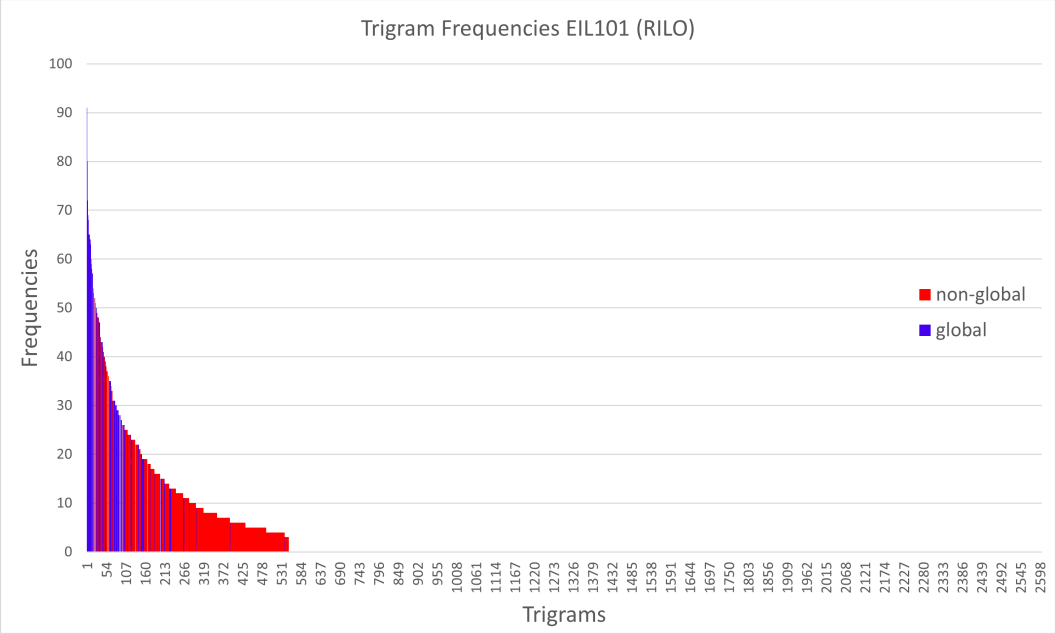



**(a)** EIL101 Top 200 Bigrams ACO-S1

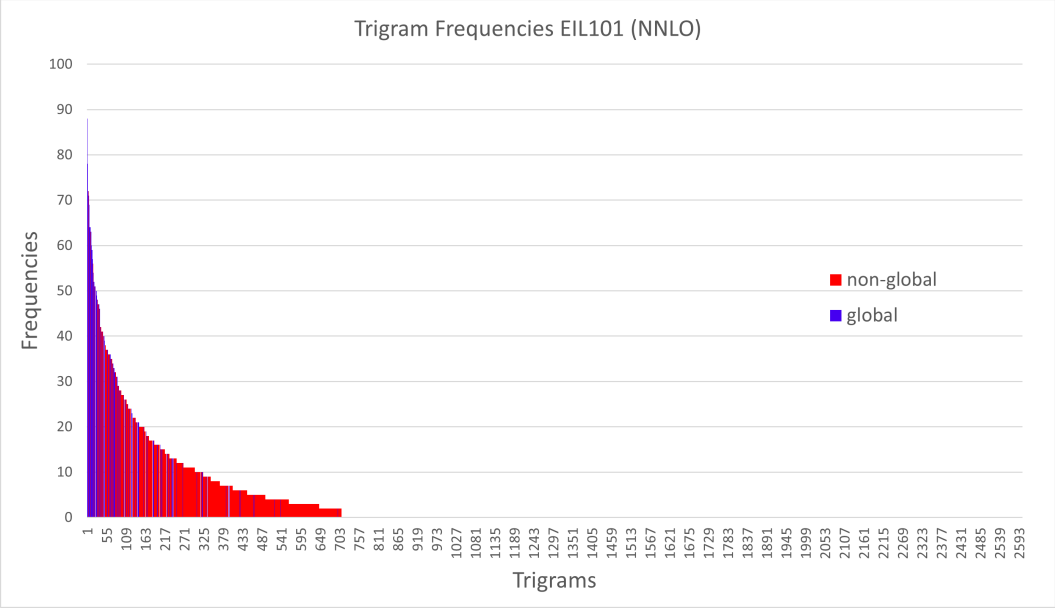



**(b)** EIL101 Top 200 Bigrams ACO-LO

**Figure 4.10:** Top 200 EIL101 Bigrams (a) ACO-S1 (b) ACO-LO
Note: blue edges are globally optimal edges while red are locally optimal edges. The edge with teal vertices is the first ranked edge.

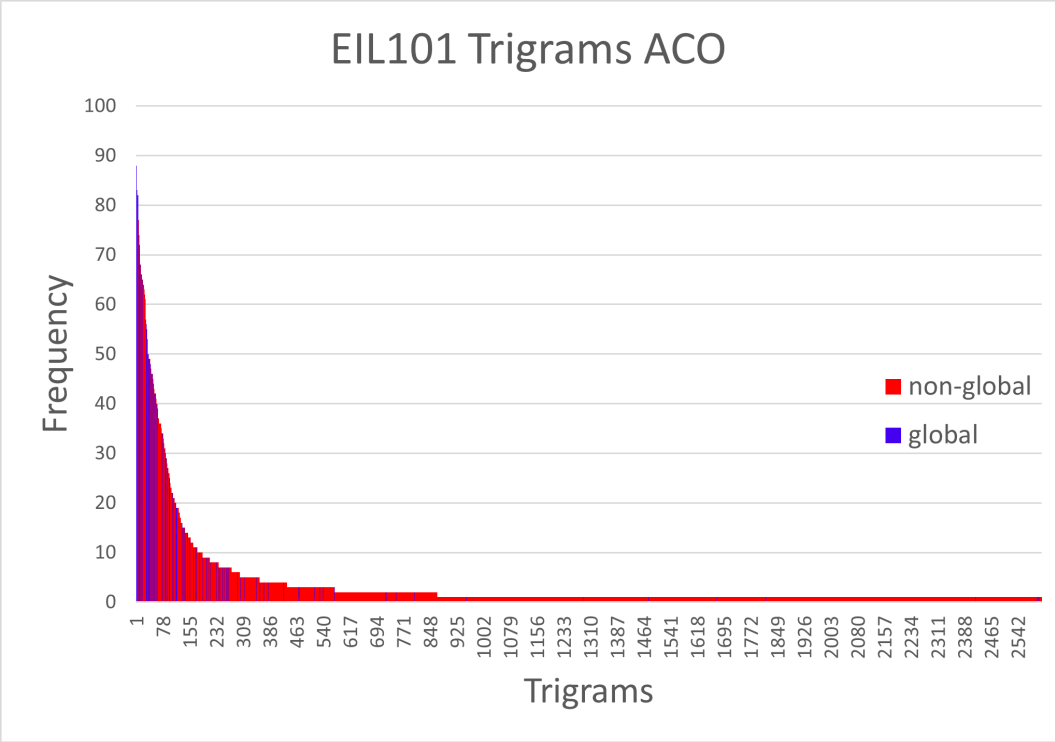



**(a)** EIL101 Trigrams RILO

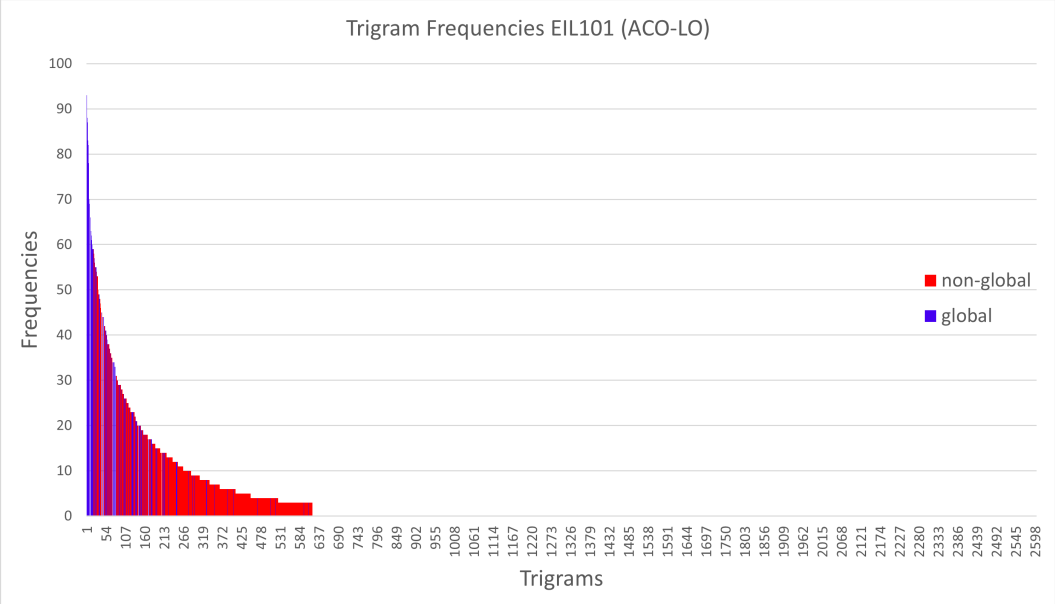



**(b)** EIL101 Trigrams NNLO

**Figure 4.11:** Trigram Frequencies for the EIL101 vs. the Number of Trigrams (a) RILO (b) NNLO




**(a)** EIL101 Trigrams ACO-S1




**(b)** EIL101 Trigrams ACO-LO

**Figure 4.12:** Trigram Frequencies for the EIL101 vs. the Number of Trigrams (a) ACO-S1 (b) ACO-LO

# Chapter 5

# Discussion and Conclusion

## 5.1 Summary

The Traveling Salesman Problem remains an interesting problem to solve with approximate solvers and exact solvers alike, especially considering the many applications. The challenge is that as the problem size continues to grow, the solvers have a tendency to get more and more inefficient. Finding ways to narrow the search space and trim out suboptimal results quickly can speed up this process. As we learn more about methods to generate viable solutions, we can continue to improve the results and learn more about ways to tackle this and other NP-Hard problems.

In this work, we focus on tour construction methods combined with 2-opt for tour improvement to generate local optima. The ATT48 and EIL101 were examined in this thesis in order to compare various methods of tour construction algorithms improved with 2-opt. We wanted to better understand the role unbiased and biased sampling methods play on the frequency of global edges within the set of local optima. Our hypothesis was that the unbiased method, randomly initialized local optima, would be better at sampling global edges than the biased methods, nearest neighbors and ACO. This results in a greater number of global edges in each size of n-gram and seeing the global edges represented at higher frequencies within the sample of 100 local optima.

After analyzing the results, RILO captured the global edges at higher frequencies as seen in the tables and plots of n-grams for the ATT48 and the EIL101. While RILO did not always gather more of the global edges in each instance, the RILO samples consistently gathered all of the global edges at higher frequencies. There is greater representation of global edges in the samples for both problem instances, and they were finding all of the global edges earlier in the

frequency tables than the other methods. Thus, our hypothesis that RILO would be better at gathering globally optimal edges at higher frequencies is supported by the data.

However, for the ATT48, the ACO with 2-opt was able to get the global optimum by iteration 312. When we noticed that the costs were getting close to the optimal, we ran the ACO algorithm further to identify where it was able to reach the global optimum once 2-opt was applied to improve the sampled tours. While this was after the range of static set selected for comparison (200-300), it is important to note that none of the other methods were able to find the global maximum within their samples either. For the EIL101, none of these tour construction methods were able to find the global optimum within the current parameters. Note that the ACO did more work, though. For the static sample of 100 permutations, there were 100 ants generating 100 permutations at each iteration and only the best at each iteration was kept in the range of 200-300 iterations.

While not a metric represented in the tables, the downside to randomly initialized tours is that they take longer to improve with 2-opt than nearest neighbors and the ant colony optimization tours. With nearest neighbors especially, the tour is being built with the intention of choosing shorter edges, so there is often less room for improvement. However, it was evident with the ant colony sample before it was improved that there were clusters of high frequency non-global edges at the top, meaning that the bias can also have a negative impact if non-global edges are being prioritized over global edges. While modifying the parameters can change the results, more iterations of the ant colony algorithm do not necessarily lead to dramatically improved results. The ants can get trapped in local optima, and adding more iterations may not allow them to find better local optima or the global optimum. When running more iterations of the EIL101, the best result, or lowest cost, with ACO and 2-opt for 300-400 iterations was 630, the best result for 500-600 was 628, and the best result for 900-1000 was 625. Interestingly, from 1000-1100 the best result was 627. With our rounding, the cost of the global optimum is 623, so these results get close, but they do not reach the global optimum.

If the goal is to introduce diversity, randomly initialized tours are a likely choice due to the unbiased nature. If the goal is to reconstruct the global optimum, the unbiased method is also superior as the globally optimal edges are higher ranked in the frequency tables. However, ACO with 2-opt was able to get the global optimum with the ATT48 and the parameters we used.

## 5.2 Future Work

Though this contributes to the discussion on the effectiveness of the ant colony algorithm, there is more to explore in this area. There are other tour construction methods that could be compared, and we only used 2-opt to improve these methods. There are different versions of 2-opt that can run faster, and there are other ways to generate local optima. There are also many variations of the ant colony algorithm, and the associated parameters could be optimized more.

Similarly, other TSP problems could be explored. We chose two city instances, but the results could differ if we examined cluster or printed circuit board problems, for example. For nearest neighbors especially, the cluster problems are interesting to handle. It would be enlightening to see how ant colony algorithms handle cluster problems and determine if ACO algorithms have similar struggles as nearest neighbor algorithms with the grouped nodes.

Likewise, it may be possible to improve the ant colony algorithm using these frequencies, which was our second hypothesis. Rather than using the pheromone matrix, the frequencies between edges could be input so that higher frequency edges are selected more often, which would mean selecting more globally optimal edges. This could lead to better results with fewer iterations of the ant colony algorithm. With the pheromones, their order and intensity depends entirely on the whims of the artificial ants and the order they happen to occur in, so using sampled frequencies could lead to more predictable results. The downside could be less diversity if the frequencies lead the ants down the same paths repeatedly, but if they lead the ants to better paths and the global optimum consistently or faster, then the frequencies could be a significant improvement to the ant colony optimization algorithms.

We believe that there are interesting uses for the frequency results. Using the high frequency edges, it is possible to reconstruct a large portion of the global optimum. In Figure 5.1, the high frequency 5-grams from RILO and ACO-LO are used to trace a possible tour. Globally optimal edges are represented in the red font while non-global edges are black. This is also shown on the first column of each section with 1's for global 5-grams and 0's for non-global. We started with the highest ranked edge and worked outward. One interesting part of using the frequencies like this is that you can trace where the discrepancies occur. Across the diagonal, if they are in agreement you will see the same number repeated. Examples of this are shown with bold outlines. However, as 5-grams keep getting added outward from the highest ranked 5-gram, they start to disagree, and that is shown on the diagonal as well. For example, in the RILO column there is a discrepancy that starts two rows above the 46 diagonal with 12, 31, 31, 31. This makes sense because the edge beginning with 12 is a non-global edge. Gaps in the chart show where the tours are not able to connect cleanly. The RILO column has 28 global 5-grams and 19 non-globlal while the ACO-LO column has the opposite. In the sections with many red edges, it is evident how the correct results are self-consistent. The red diagonals all agree on the order of cities within the edges.

By using this frequency information, there are likely ways to narrow the search space. If we can quickly identify areas where the results are self-consistent, then we can narrow down where we want to search to focus on the areas that disagree. Instead of running a tour improvement algorithm like 2-opt over the entire tour, we could hone in on the sections that disagree, and only improve the parts that need to be improved, which could make the process to generate optima much more efficient. With larger problems especially, narrowing down the possible search space would be an effective way to get competitive solutions efficiently.

| 2-OPT | | | | | | | | | | | ANT-CO | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | | | | |
| 0 | 24 | 13 | 12 | 20 | 46 | | | | | | 0 | 10 | 22 | 2 | 21 | 15 | | | | | |
| 0 | 38 | 24 | 13 | 12 | 20 | | | | | | 0 | 11 | 10 | 22 | 2 | 21 | | | | | |
| 0 | | | | | | | | | | | 0 | 14 | 11 | 10 | 22 | 2 | | | | | |
| 0 | 40 | 15 | 21 | 0 | 7 | | | | | | 0 | 39 | 14 | 11 | 10 | 22 | | | | | |
| 0 | 15 | 21 | 2 | 33 | 40 | | | | | | 0 | 0 | 39 | 14 | 11 | 10 | | | | | |
| 0 | 21 | 2 | 22 | 10 | 11 | | | | | | 0 | 7 | 0 | 39 | 14 | 11 | | | | | |
| 0 | 2 | 22 | 10 | 11 | 14 | | | | | | 0 | | | | | | | | | | |
| 0 | 22 | 10 | 11 | 14 | 39 | | | | | | 0 | 28 | 1 | 41 | 25 | 3 | | | | | |
| 0 | 10 | 11 | 14 | 39 | 2 | | | | | | 0 | 4 | 28 | 1 | 25 | 3 | | | | | |
| 1 | 11 | 14 | 39 | 8 | 0 | | | | | | 0 | | | | | | | | | | |
| 0 | | | | | | | | | | | 0 | 31 | 38 | 20 | 12 | 46 | | | | | |
| 1 | 41 | 4 | 47 | 38 | 31 | | | | | | 0 | 46 | 20 | 12 | 13 | 24 | | | | | |
| 1 | 4 | 47 | 38 | 31 | 20 | | | | | | 0 | 20 | 12 | 13 | 24 | 38 | | | | | |
| 1 | 47 | 38 | 31 | 20 | 46 | | | | | | 0 | 12 | 13 | 24 | 38 | 31 | | | | | |
| 1 | 38 | 31 | 20 | 46 | 19 | | | | | | 0 | 13 | 24 | 38 | 14 | 11 | | | | | |
| 0 | 12 | 20 | 46 | 19 | 32 | | | | | | 0 | 24 | 38 | 31 | 23 | 41 | | | | | |
| 1 | 20 | 46 | 19 | 32 | 45 | | | | | | 0 | 38 | 31 | 23 | 9 | 44 | | | | | |
| 1 | 46 | 19 | 32 | 45 | 35 | | | | | | 0 | 31 | 23 | 9 | 44 | 34 | | | | | |
| 1 | 19 | 32 | 45 | 35 | 29 | | | | | | 1 | 23 | 9 | 44 | 34 | 3 | | | | | |
| 1 | 32 | 45 | 35 | 29 | 42 | | | | | | 1 | 9 | 44 | 34 | 3 | 25 | | | | | |
| 1 | 45 | 35 | 29 | 42 | 16 | | | | | | 1 | 44 | 34 | 3 | 25 | 1 | | | | | |
| 1 | 35 | 29 | 42 | 16 | 26 | | | | | | 1 | 34 | 3 | 25 | 1 | 28 | | | | | |
| 1 | 29 | 42 | 16 | 26 | 18 | | | | | | 0 | 3 | 25 | 1 | 28 | 4 | | | | | |
| 1 | 42 | 16 | 26 | 18 | 36 | | | | | | 0 | 41 | 1 | 28 | 4 | 47 | | | | | |
| 1 | 16 | 26 | 18 | 36 | 5 | | | | | | 0 | 1 | 28 | 4 | 47 | 33 | | | | | |
| 1 | 26 | 18 | 36 | 5 | 27 | | | | | | 0 | 28 | 4 | 47 | 33 | 40 | | | | | |
| 1 | 18 | 36 | 5 | 27 | 6 | | | | | | 0 | 4 | 47 | 33 | 40 | 15 | | | | | |
| 1 | 36 | 5 | 27 | 6 | 17 | | | | | | 0 | 47 | 33 | 40 | 15 | 21 | | | | | |
| 1 | 5 | 27 | 6 | 17 | 43 | | | | | | 0 | 33 | 40 | 15 | 21 | 0 | | | | | |
| 1 | 27 | 6 | 17 | 43 | 30 | | | | | | 0 | 40 | 15 | 21 | 0 | 7 | | | | | |
| 1 | 6 | 17 | 43 | 30 | 37 | | | | | | 0 | 15 | 21 | 0 | 7 | 8 | | | | | |
| 0 | 17 | 43 | 30 | 37 | 8 | | | | | | 0 | 21 | 0 | 7 | 8 | 37 | | | | | |
| 0 | 43 | 30 | 37 | 8 | 7 | | | | | | 0 | 0 | 7 | 8 | 37 | 30 | | | | | |
| 0 | 30 | 37 | 8 | 7 | 0 | | | | | | 0 | 7 | 8 | 37 | 30 | 43 | | | | | |
| 0 | 37 | 8 | 7 | 0 | 15 | | | | | | 0 | 8 | 37 | 30 | 43 | 17 | | | | | |
| 0 | 8 | 7 | 0 | 15 | 21 | | | | | | 1 | 37 | 30 | 43 | 17 | 6 | | | | | |
| 0 | 7 | 0 | 15 | 21 | 2 | | | | | | 1 | 30 | 43 | 17 | 6 | 27 | | | | | |
| 0 | 40 | 15 | 21 | 0 | 7 | | | | | | 1 | 43 | 17 | 6 | 27 | 5 | | | | | |
| 0 | 15 | 21 | 2 | 33 | 40 | | | | | | 1 | 17 | 6 | 27 | 5 | 36 | | | | | |
| 0 | 21 | 2 | 33 | 40 | 28 | | | | | | 1 | 6 | 27 | 5 | 36 | 18 | | | | | |
| 0 | 2 | 33 | 40 | 28 | 1 | | | | | | 1 | 27 | 5 | 36 | 18 | 26 | | | | | |
| 0 | 47 | 4 | 28 | 1 | 41 | | | | | | 1 | 5 | 36 | 18 | 26 | 16 | | | | | |
| 0 | 40 | 28 | 1 | 25 | 3 | | | | | | 1 | 36 | 18 | 26 | 16 | 42 | | | | | |
| 1 | 28 | 1 | 25 | 3 | 34 | | | | | | 1 | 18 | 26 | 16 | 42 | 29 | | | | | |
| 1 | 1 | 25 | 3 | 34 | 44 | | | | | | 1 | 26 | 16 | 42 | 29 | 35 | | | | | |
| 1 | 25 | 3 | 34 | 44 | 9 | | | | | | 1 | 16 | 42 | 29 | 35 | 45 | | | | | |
| 1 | 3 | 34 | 44 | 9 | 23 | | | | | | 1 | 42 | 29 | 35 | 45 | 32 | | | | | |
| 1 | 34 | 44 | 9 | 23 | 41 | | | | | | 1 | 29 | 35 | 45 | 32 | 19 | | | | | |
| 1 | 44 | 9 | 23 | 41 | 4 | | | | | | 1 | 35 | 45 | 32 | 19 | 46 | | | | | |
| 1 | 9 | 23 | 41 | 4 | 47 | | | | | | 1 | 45 | 32 | 19 | 46 | 20 | | | | | |
| 1 | 23 | 41 | 4 | 47 | 6 | | | | | | 0 | 32 | 19 | 46 | 20 | 12 | | | | | |
| 28 | | | | | | | | | | | 19 | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | |
| YES | 28 | | | | | | | | | | YES | 19 | | | | | | | | | |
| NO | 19 | | | | | | | | | | NO | 28 | | | | | | | | | |

**Figure 5.1:** Using High Frequency 5-grams to Build A Tour (a) RILO (b) ACO-LO

# Bibliography


[1] Keld Helsgaun. An effective implementation of the lin–kernighan traveling salesman heuristic. *European Journal of Operational Research*, 126(1):106–130, 2000.

[2] D. Applegate, R. Bixby, V. Chvatal, and B. Cook. Finding cuts in the tsp (a preliminary report). Technical report, 1995.

[3] Paul McMenemy, Nadarajen Veerapen, Jason Adair, and Gabriela Ochoa. Rigorous performance analysis of state-of-the-art tsp heuristic solvers. *European Conference on Evolutionary Computation in Combinatorial Optimization*, page 99–114, Apr 2019.

[4] William J. Cook. *In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation*. Princeton University Press, 2014.

[5] Gregory Gutin, Anders Yeo, and Alexey Zverovich. Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the tsp. *Discrete Applied Mathematics*, 117(1):81–86, 2002.

[6] Sourabh Joshi and Sarabjit Kaur. Nearest neighbor insertion algorithm for solving capacitated vehicle routing problem. In *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 86–88, 2015.

[7] Janez Brest and Janez Zerovnik. A heuristic for the asymmetric traveling salesman problem. *The 6th Metaheuristics International Conference (MIC2005)*, Aug 2005.

[8] Daniel J. Rosenkrantz, Richard E. Stearns, and Philip M. Lewis, II. An analysis of several heuristics for the traveling salesman problem. *SIAM Journal on Computing*, 6(3):563–581, Sep 1977.

[9] Marco Dorigo and Luca Maria Gambardella. Ant colonies for the travelling salesman problem. *Biosystems*, 43(2):73–81, 1997.

[10] G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, 1959.

[11] Kris Braekers, Katrien Ramaekers, and Inneke Van Nieuwenhuyse. The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99:300–313, 2016.

[12] John Berroa. Ant colony optimization. https://github.com/johnberroa/Ant-Colony-Optimization, 2018.

[13] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm intelligence: From natural to Artificial Systems*. Oxford Univ. Press, 1999.

[14] Stephen Gilmour and Mark Dras. Understanding the pheromone system within ant colony optimization. In Shichao Zhang and Ray Jarvis, editors, *AI 2005: Advances in Artificial Intelligence*, pages 786–789, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[15] Hui Yu. Optimized ant colony algorithm by local pheromone update. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, 12, 09 2014.

[16] Swetha Varadarajan, Darrell Whitley, and Gabriela Ochoa. Why many travelling salesman problem instances are easier than you think. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, GECCO '20, page 254–262, New York, NY, USA, 2020. Association for Computing Machinery.

[17] Shi Cheng, Bin Liu, T. O. Ting, Quande Qin, Yuhui Shi, and Kaizhu Huang. Survey on data science with population-based algorithms. *Big Data Analytics*, 1(1), Jul 2016.

[18] Enmin Zhu, Jianjie Zhang, Jijie Yan, Kongyang Chen, and Chongzhi Gao. N-gram malgan: Evading machine learning detection via feature n-gram. *Digital Communications and Networks*, 2021.

# Appendix A

# Additional Tables and Plots

**Table A.1:** Table of All of the Global 5-grams Collected by ACO-S1 for the ATT48 (only 22 were found)

| **ATT48 Global 5-grams ACO-S1** | | | | | | |
|---|---|---|---|---|---|---|
| **rank** | **edge** | | | | | |
| 35 | 18 | 26 | 16 | 42 | 29 | |
| 36 | 2 | 21 | 15 | 40 | 33 | |
| 37 | 20 | 46 | 19 | 32 | 45 | |
| 38 | 36 | 18 | 26 | 16 | 42 | |
| 39 | 6 | 17 | 43 | 30 | 37 | |
| 42 | 27 | 6 | 17 | 43 | 30 | |
| 55 | 5 | 36 | 18 | 26 | 16 | |
| 59 | 22 | 2 | 21 | 15 | 40 | |
| 65 | 2 | 22 | 13 | 24 | 12 | |
| 83 | 5 | 27 | 6 | 17 | 43 | |
| 84 | 26 | 16 | 42 | 29 | 35 | |
| 90 | 26 | 18 | 36 | 5 | 27 | |
| 143 | 6 | 27 | 5 | 36 | 18 | |
| 144 | 17 | 6 | 27 | 5 | 36 | |
| 277 | 37 | 7 | 0 | 8 | 39 | |
| 660 | 13 | 22 | 2 | 21 | 15 | |
| 828 | 0 | 8 | 39 | 14 | 11 | |
| 1000 | 19 | 32 | 45 | 35 | 29 | |
| 1001 | 35 | 45 | 32 | 19 | 46 | |
| 1699 | 10 | 12 | 24 | 13 | 22 | |
| 1700 | 11 | 10 | 12 | 24 | 13 | |
| 1701 | 14 | 11 | 10 | 12 | 24 | |

**Table A.2:** Table of Top 40 5-grams for the ACO-S1 for the ATT48

| **ATT48 Top 40 5-grams ACO-S1** | | | | | |
|---|---|---|---|---|---|
| **edge** | | | | | **freq** |
| 38 | 31 | 23 | 9 | 41 | 98 |
| 23 | 9 | 41 | 44 | 34 | 98 |
| 9 | 41 | 44 | 34 | 25 | 98 |
| 3 | 25 | 34 | 44 | 41 | 98 |
| 31 | 23 | 9 | 41 | 44 | 97 |
| 1 | 3 | 25 | 34 | 44 | 97 |
| 28 | 1 | 3 | 25 | 34 | 97 |
| 4 | 28 | 1 | 3 | 25 | 96 |
| 3 | 1 | 28 | 4 | 47 | 96 |
| 28 | 4 | 47 | 33 | 40 | 70 |
| 4 | 47 | 33 | 40 | 15 | 70 |
| 0 | 21 | 15 | 40 | 33 | 70 |
| 7 | 0 | 21 | 15 | 40 | 70 |
| 1 | 28 | 4 | 47 | 33 | 69 |
| 8 | 7 | 0 | 21 | 15 | 67 |
| 21 | 0 | 7 | 8 | 37 | 66 |
| 21 | 15 | 40 | 33 | 47 | 65 |
| 0 | 7 | 8 | 37 | 30 | 55 |
| 9 | 23 | 31 | 38 | 20 | 49 |
| 9 | 23 | 31 | 38 | 24 | 44 |
| 13 | 24 | 38 | 31 | 23 | 30 |
| 30 | 45 | 32 | 19 | 46 | 28 |
| 30 | 43 | 17 | 6 | 35 | 27 |
| 6 | 17 | 43 | 30 | 45 | 27 |
| 18 | 36 | 16 | 42 | 29 | 26 |
| 17 | 43 | 30 | 45 | 32 | 26 |
| 7 | 8 | 37 | 30 | 43 | 26 |
| 26 | 18 | 36 | 16 | 42 | 25 |
| 12 | 20 | 38 | 31 | 23 | 25 |
| 19 | 32 | 45 | 30 | 43 | 24 |
| 5 | 26 | 18 | 36 | 16 | 23 |
| 8 | 37 | 30 | 43 | 17 | 23 |
| 7 | 8 | 37 | 30 | 45 | 22 |
| 8 | 37 | 30 | 45 | 32 | 21 |
| 18 | 26 | 16 | 42 | 29 | 20 |
| 2 | 21 | 15 | 40 | 33 | 20 |
| 20 | 46 | 19 | 32 | 45 | 19 |
| 36 | 18 | 26 | 16 | 42 | 19 |
| 6 | 17 | 43 | 30 | 37 | 19 |
| 27 | 35 | 6 | 17 | 43 | 18 |

**Table A.3:** Table Top 20, Bottom 20 Global 5-grams for ACO-S1 for the EIL101

| **EIL 101 Top 20, Bottom 20 Global 5-grams ACO** | | | | | |
|---|---|---|---|---|---|
| rank | edge | | | | |
| 11 | 1 | 56 | 14 | 42 | 41 |
| 19 | 4 | 83 | 16 | 44 | 7 |
| 30 | 16 | 83 | 4 | 59 | 82 |
| 34 | 0 | 49 | 75 | 76 | 2 |
| 44 | 68 | 0 | 49 | 75 | 76 |
| 45 | 70 | 64 | 34 | 33 | 77 |
| 46 | 36 | 97 | 99 | 90 | 84 |
| 47 | 26 | 68 | 0 | 49 | 75 |
| 55 | 49 | 75 | 76 | 2 | 78 |
| 56 | 44 | 16 | 83 | 4 | 59 |
| 63 | 38 | 66 | 22 | 55 | 74 |
| 67 | 49 | 0 | 68 | 26 | 100 |
| 69 | 71 | 73 | 21 | 40 | 74 |
| 75 | 33 | 34 | 64 | 70 | 65 |
| 80 | 55 | 74 | 40 | 21 | 73 |
| 103 | 42 | 14 | 56 | 1 | 86 |
| 119 | 0 | 68 | 26 | 100 | 52 |
| 122 | 17 | 82 | 59 | 4 | 83 |
| 147 | 60 | 84 | 90 | 99 | 97 |
| 169 | 19 | 65 | 70 | 64 | 34 |
| | | | | | |
| 967 | 88 | 5 | 95 | 98 | 92 |
| 1938 | 51 | 88 | 5 | 95 | 98 |
| 2125 | 29 | 19 | 65 | 70 | 64 |
| 2126 | 69 | 29 | 19 | 65 | 70 |
| 2297 | 39 | 57 | 12 | 93 | 94 |
| 2298 | 57 | 12 | 93 | 94 | 96 |
| 2332 | 64 | 34 | 33 | 77 | 80 |
| 2445 | 32 | 78 | 2 | 76 | 75 |
| 2483 | 13 | 43 | 37 | 85 | 15 |
| 2484 | 43 | 37 | 85 | 15 | 60 |
| 2485 | 37 | 85 | 15 | 60 | 84 |
| 2968 | 3 | 38 | 66 | 22 | 55 |
| 3286 | 40 | 74 | 55 | 22 | 66 |
| 3321 | 24 | 54 | 53 | 23 | 28 |
| 4070 | 57 | 39 | 20 | 72 | 71 |
| 4248 | 25 | 27 | 52 | 100 | 26 |
| 4249 | 27 | 52 | 100 | 26 | 68 |
| 5277 | 53 | 23 | 28 | 67 | 79 |
| 5354 | 5 | 95 | 98 | 92 | 58 |
| 5355 | 91 | 58 | 92 | 98 | 95 |

**Table A.4:** Table of the Top 40 5-grams for ACO-S1 for the EIL101

| **EIL101 Top 40 5-grams ACO** | | | | | |
|---|---|---|---|---|---|
| **edge** | | | | | **freq** |
| 20 | 71 | 73 | 21 | 40 | 56 |
| 42 | 14 | 56 | 1 | 72 | 54 |
| 21 | 73 | 71 | 20 | 72 | 54 |
| 1 | 72 | 20 | 71 | 73 | 53 |
| 14 | 56 | 1 | 72 | 20 | 52 |
| 44 | 7 | 81 | 47 | 46 | 51 |
| 56 | 1 | 72 | 20 | 71 | 51 |
| 95 | 98 | 58 | 91 | 96 | 47 |
| 16 | 44 | 7 | 81 | 47 | 45 |
| 33 | 34 | 70 | 64 | 65 | 44 |
| 1 | 56 | 14 | 42 | 41 | 43 |
| 64 | 70 | 34 | 33 | 77 | 43 |
| 94 | 96 | 91 | 58 | 98 | 42 |
| 28 | 23 | 54 | 24 | 38 | 41 |
| 19 | 65 | 64 | 70 | 34 | 41 |
| 81 | 7 | 44 | 16 | 83 | 39 |
| 5 | 93 | 94 | 96 | 91 | 39 |
| 17 | 88 | 5 | 93 | 94 | 37 |
| 4 | 83 | 16 | 44 | 7 | 36 |
| 58 | 91 | 96 | 94 | 93 | 35 |
| 70 | 34 | 33 | 77 | 78 | 35 |
| 22 | 66 | 38 | 24 | 54 | 35 |
| 13 | 37 | 43 | 15 | 60 | 35 |
| 56 | 14 | 42 | 41 | 86 | 34 |
| 22 | 40 | 21 | 73 | 71 | 34 |
| 23 | 54 | 24 | 38 | 66 | 33 |
| 88 | 5 | 93 | 94 | 96 | 32 |
| 15 | 43 | 37 | 13 | 42 | 32 |
| 14 | 42 | 13 | 37 | 43 | 32 |
| 16 | 83 | 4 | 59 | 82 | 31 |
| 37 | 13 | 42 | 14 | 56 | 30 |
| 3 | 54 | 24 | 38 | 66 | 29 |
| 55 | 22 | 40 | 21 | 73 | 28 |
| 0 | 49 | 75 | 76 | 2 | 28 |
| 29 | 19 | 65 | 64 | 70 | 28 |
| 12 | 86 | 41 | 42 | 14 | 27 |
| 1 | 56 | 14 | 42 | 13 | 27 |
| 24 | 54 | 23 | 28 | 78 | 27 |
| 28 | 78 | 77 | 33 | 34 | 26 |
| 23 | 28 | 78 | 77 | 33 | 26 |

# Appendix B

# License

**Colorado State University LaTeX Thesis Template**

by Elliott Forney – 2017

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.